

Chemometrie in der NMR-Spektroskopie – Signalanalyse mittels neuronaler Netze

von
Diplom-Chemiker
Marco Sielaff

Vom Fachbereich 5 – Chemie –
der Technischen Universität Berlin
zur Erlangung des akademischen Grades

Doktor der Naturwissenschaften
– Dr. rer. nat. –

genehmigte Dissertation

Promotionsausschuss :

Vorsitzender: Prof. Dr. J. Starnick
Berichter: Prof. Dr. D. Ziessow
Berichter: Prof. Dr. G. H. Findenegg
Tag der mündlichen Prüfung: 15.12.2000

Berlin 2000

D 83

Abstract

Sielaff, Marco

Chemometrie in der NMR-Spektroskopie – Signalanalyse mittels neuronaler Netze

Gegenstand der Arbeit ist die Weiterentwicklung messzeiteffizienter und fehlertoleranter Verfahren der NMR-Spektroskopie, bei denen die Antwort gekoppelter Spinsysteme $y(t)$ auf eine Anregung mit Zufallsfolgen von Hochfrequenzimpulsen $x(t)$ gemessen wird. Der Zusammenhang beider Zeitfunktionen ist theoretisch durch eine unendliche Funktionalreihe $y(t) = \Phi[x(t)]$ gegeben, in der Φ das untersuchte Spinsystem beschreibt.

Startpunkt der Arbeit ist der seit einigen Jahren bekannte Lösungsansatz, die Funktionalreihe Φ durch ein neuronales Netz (NN) zu approximieren. Seine Gewichte werden gemäß der verwendeten Anregung und gemessenen Spinantwort mit einem Algorithmus ermittelt (Training) und charakterisieren das untersuchte System. Die gesuchte spektrale NMR-Information entsteht rechnerisch durch Abfrage des trainierten NN mit der entsprechenden NMR-Anregung oder alternativ direkt aus den NN -Gewichten (NN -NMR).

Die NN -NMR ist messzeiteffizient (geringe Verweilzeit der Messprobe im Magneten) und tolerant gegenüber Messparameterfehlern, führt jedoch zu erheblichen NN -Trainingszeiten und approximationsbedingten Informationsverlusten. Daher wurde untersucht, inwieweit diese Nachteile durch neue Netztopologien und Lernalgorithmen sowie unter Verwendung von Vorinformationen soweit minimiert werden können, dass die NN -NMR auch für größere Spinsysteme und Substanzgemische praktikabel ist.

Alternative Lernalgorithmen wurden auf ihre Tauglichkeit für die NN -NMR überprüft. Erfolgversprechend erwies sich insbesondere eine Netztopologie, die die aus dem gemessenen 1D-NMR-Spektrum bekannten Resonanzfrequenzen integriert. Dieser Ansatz erlaubt es, die Größe des verwendenden NN zu reduzieren (schnelleres NN -Training) und ermöglicht die Untersuchung von Molekülen mit vielen gekoppelten Spins oder Substanzgemischen mit einem einzigen NN . Bei dieser Netzart werden die gesuchten Spinkonnektivitäten von der aus den Messdaten ermittelten NN -Gewichtsmatrix direkt angezeigt und müssen nicht aus dem gemessenen 2D-Spektrum der konventionellen NMR-Spektroskopie per Inspektion oder Mustererkennungsprogramm abgelesen werden.

Simulationsrechnungen und Experimente demonstrieren den erreichten Fortschritt der NN -NMR. Dies betrifft insbesondere die Verwendung komplexer NN -Gewichte und die schnelle Identifizierung und Erkennung von Teilspinsystemen mit gleichem Koppelungsmuster bei Vorgabe ihrer bekannten Resonanzfrequenzen. Weiter wird gezeigt, dass bei der NN -NMR bereits während des Messprozesses Strukturisomere für eine gegebene Bruttoformel ausgeschlossen werden können. Hierzu werden für alle von einem Strukturgenerator erzeugten und von vornherein nicht ausschließbaren Isomeren i die zu erwartenden NN -Zeitfunktionen $y_i(t)$ vorausberechnet. Die Korrelation von $y_i(t)$ mit den eingehenden Messdaten $y_{exp}(t)$ macht es bereits vor dem NN -Training möglich, nichtzutreffende Isomere auszusondern. Ergibt sich eine hinreichende Übereinstimmung mit einer Zielstruktur, ggf. unter Einbeziehung der schnellen Erkennung charakteristischer Teilstrukturen, so kann auf das rechenintensive Gesamt- NN -Training verzichtet werden.

Inhaltsverzeichnis

Inhaltsverzeichnis	I
Abbildungsverzeichnis	III
Tabellenverzeichnis	V
Symbolverzeichnis	VII
1 Einleitung	1
2 Theoretische Grundlagen	3
2.1 Theorie der mehrdimensionalen NMR-Spektroskopie	3
2.1.1 Quantenmechanische Beschreibung	3
2.1.2 Eindimensionale NMR-Spektroskopie	4
2.1.3 Mehrdimensionale NMR-Spektroskopie	5
2.2 Systemtheorie	7
2.2.1 Charakterisierung	7
2.2.2 Lineare Systeme	8
2.2.3 Nichtlineare Systeme	8
2.2.4 NMR-Systeme	9
2.3 Neuronale Netze	11
2.3.1 Grundlagen	11
2.3.2 Neuronale Netze in der NMR-Signalanalyse (<i>NN-NMR</i>)	12
2.4 Methodik der <i>NN-NMR</i>	15
2.4.1 Messung der Spinsystemdaten	15
2.4.2 Netztraining	16
2.4.3 Generierung von Zeitdaten	17
2.5 Spezielle Selektionsverfahren	19
2.5.1 Digitale Filterung	19
2.5.2 Verwendung von Vorinformationen	19
2.6 Bewertungskriterium für alternative <i>NN</i> -Verfahren	21
3 Programme und Apparaturen	23
3.1 Verwendete Programme für die NMR-Simulationen	23
3.1.1 <i>SPECSI</i>	23
3.1.2 <i>GAMMA</i>	24
3.1.3 <i>ACD/HNMR</i>	24
3.2 Verwendete Programme für neuronale Netze	24

3.2.1	Spezielle Programmcodes zur NMR-Signalanalyse	24
3.2.2	<i>SNNS</i>	25
3.3	Experimenteller Teil	25
3.3.1	Benutzte NMR-Apparatur	25
3.3.2	Messung	26
4	Arbeiten und Resultate	27
4.1	Entwicklungen zur <i>NN</i> -Methodik	27
4.1.1	Minimierung der Netzgröße	27
4.1.2	Berücksichtigung der zeitlichen Ableitung von Anregung und Antwort ..	31
4.1.3	Radiale Basisfunktionsnetze	34
4.1.4	Der <i>resilient propagation</i> -Algorithmus	35
4.1.5	Der <i>backpercolation</i> -Algorithmus	37
4.1.6	Der <i>cascade-correlation</i> -Algorithmus	38
4.2	Vorgabe von Resonanzfrequenzen	40
4.2.1	Trennung von überlagerten Spinsystemen ohne Überlappung	40
4.2.2	Trennung von überlagerten Spinsystemen mit überlappenden Signalen ..	43
4.2.3	Quadraturdetektion	47
4.2.4	Spinsysteme höherer Ordnung	51
4.2.5	Auswahlkriterien für bekannte Summenformeln	53
4.2.6	Konnektivitätsbestimmung aus experimentellen Daten	57
5	Zusammenfassung	61
	Literatur	63
	Anhänge	67
A.1	Verwendete NMR-Parameter	67
A.1.1	Verwendetes Spinsystem zur Überprüfung des <i>pruning</i> -Verfahrens	67
A.1.2	Verwendete Spinsysteme zur Überprüfung des <i>RPROP</i> -Algorithmus	67
A.1.3	Konnektivitätsbestimmung zur Trennung von überlagerten Spinsysteme- men ohne Überlappung	67
A.1.4	Konnektivitätsbestimmung zur Trennung von überlagerten Spinsysteme- men mit überlappenden Signalen	68
A.1.5	Konnektivitätsbestimmung von 4-Aminobuttersäure	68
A.1.6	Konnektivitätsbestimmung von Phenol mit einem Kontrollsignal	68
A.1.7	Spinsysteme mit der Summenformel C_4H_6	68
A.2	Optimierung der Programme	70
A.2.1	Optimierung durch Compilerschalter	70
A.2.2	Optimierung des Quellcodes	70
A.2.3	Verteilung der Abläufe auf mehrere Rechnersysteme	72
A.3	<i>RPROP</i> -Algorithmus für die NMR-Signalanalyse	73

Abbildungsverzeichnis

2.1	Aufnahme eines Zeitsignals in einem 1D-Experiment.	4
2.2	Aufnahme eines <i>COSY</i> -Experiments.	6
2.3	Abbildung einer Anregung $x(t)$ auf eine Antwort $y(t)$	7
2.4	Diagonal- und Parallelschnitt des Kerns $K_3(\omega_1, \omega_2, \omega_3)$	10
2.5	Kaskade zweier Systeme zur Beschreibung von Spinsystemen ohne skalare Kopplung.	11
2.6	Neuronales Netz mit <i>feedforward</i> -Topologie.	13
2.7	Ausschnitt aus einem <i>feedforward</i> -Netz zur Funktionalapproximierung in der NMR-Signalanalyse.	13
2.8	Generierung von <i>nD</i> -NMR-Datensätzen (Zeitdomäne) für eine gegebene NMR-Probe.	16
2.9	Schema zum Training des neuronalen Netzes.	17
2.10	Modifiziertes neuronales Netz zur Berücksichtigung von Vorinformationen.	20
3.1	Strukturformel des 1-Chlor-2,4-dinitrobenzol.	26
4.1	Verkleinerung der Netztopologie nach dem <i>pruning</i> -Verfahren.	28
4.2	Korrelationen eines AX+P-Spinsystems und Rechenzeitreduktion nach dem <i>pruning</i> -Verfahren.	29
4.3	2D-Schnitt für ein <i>pruning</i> -Netz eines AX+P-Spinsystems.	30
4.4	Korrelationen eines ABC+P-Spinsystems und Rechenzeitreduktion nach dem <i>pruning</i> -Verfahren.	31
4.5	Ausschnitt aus einer neuronalen Anregungssequenz und seiner Ableitung.	33
4.6	Neuronales Netz mit Einsatz der Ableitungsfunktion.	33
4.7	2D-Spektrum für ein AX+P-Spinsystems, das mit dem <i>RPROP</i> -Algorithmus trainiert wurde.	37
4.8	1D-Spektrum eines unbekanntes Beispielspinsystems.	40
4.9	2D-Spektrum von mehreren Spinsystemen, sowie ein 1D-Schnitt durch das 2D-Spektrum.	42
4.10	1D-Spektrum eines Beispielspinsystems mit Koppelungsschemata.	43
4.11	2D-Spektrum eines unbekanntes Systems mit überlappenden Signalen.	44
4.12	2D-Spektrum von zwei AX-Spinsystemen.	45
4.13	2D-Spektrum zur Überprüfung eines AX-Teilsystems.	46
4.14	2D-Spektrum zur Überprüfung eines nicht koppelnden Spinsystems.	47
4.15	Simuliertes 1D-Spektrum in Quadratur von 4-Aminobuttersäure.	48
4.16	2D-Spektrum in Quadratur von 4-Aminobuttersäure.	50
4.17	Simuliertes 1D-Spektrum von Phenol.	51
4.18	2D-Spektrum für Phenol und einem Kontrollsignal.	52

4.19	Strukturformeln von Verbindungen mit der Summenformel C_4H_6	53
4.20	Korrelationskoeffizienten für unterschiedliche NMR-Parameter des But-1-ins. ..	54
4.21	Korrelationskoeffizienten ohne 1D-Anteil des But-1-ins.	55
4.22	Korrelationskoeffizienten für Verbindungen mit der Summenformel C_4H_6	56
4.23	2D-Teilspektrum von But-1-in.	56
4.24	Experimentelles 1D-Spektrum von 1-Chlor-2,4-dinitrobenzol.	58
4.25	Resultierendes 2D-Spektrum von 1-Chlor-2,4-dinitrobenzol bei zu wenig Vor- information.....	58
4.26	Resultierendes 2D-Spektrum von 1-Chlor-2,4-dinitrobenzol bei ausreichend Vor- information und 1D-Schnitt durch das 2D-Spektrum.	60

Tabellenverzeichnis

4.1	Gewichtsmatrix für ein überlagertes Mehrspinsystem.....	41
4.2	Gewichtsmatrix eines unbekanntes Systems mit überlappenden Signalen.	44
4.3	Gewichtsmatrix zur Trennung von überlappenden Signalen.	45
4.4	Gewichtsmatrix zur Überprüfung eines AX-Teilsystems.....	46
4.5	Gewichtsmatrix bei Vorgabe eines nicht koppelnden Spinsystems.	47
4.6	Realteil der komplexen Gewichtsmatrix für 4-Aminobuttersäure.	48
4.7	Imaginärteil der komplexen Gewichtsmatrix für 4-Aminobuttersäure.....	49
4.8	Magnitude der komplexen Gewichtsmatrix für 4-Aminobuttersäure.	49
4.9	Gewichtsmatrix für Phenol und einem Kontrollsignal.....	52
4.10	Gewichtsmatrix für But-1-in.	57
4.11	Gewichtsmatrix für 1-Chlor-2,4-dinitrobenzol.	59

Symbolverzeichnis

$\mathbf{H}(t)$	Zeitabhängiger Hamiltonoperator
\mathbf{H}_0	Zeitunabhängiger Hamiltonoperator
$\mathbf{H}_1(t)$	Zeitabhängiger Störterm des Hamiltonoperators (Hf-Impulse)
$\vec{\mathbf{I}}_l$	Spinoperator für den Kern l
\mathbf{I}_{la}	a -Komponente des Spinoperators für den Kern l , $a = x, y, z$
$\boldsymbol{\sigma}(t), \boldsymbol{\sigma}_{eq}$	Reduzierte Dichtematrix (im thermischen Gleichgewicht)
ω_1	Anregungsfunktion über das B_1 -Feld
\mathbf{Tr}	Spurbildungsoperator
Ω_l	Larmorfrequenz des Kerns l
γ, \hbar	Gyromagnetisches Verhältnis, Plancksches Wirkungsquantum
$n\mathbf{D}$	n -dimensional
$x(t), y(t)$	Anregungsfunktion, Systemantwort
$\Phi[x(t)]$	Systemübertragungsfunktional für eine zeitabhängige Anregungsfunktion $x(t)$
V_n	Volterra-Funktional der Ordnung n
$k_n(\tau_1, \dots, \tau_n)$	Volterra-Kern n -ter Ordnung
$K_n(\omega_1, \dots, \omega_n)$	Fouriertransformierte von $k_n(\tau_1, \dots, \tau_n)$
k'_2	Aus k_3 zusammengefasster zweidimensionaler Datensatz
NN	Neuronales Netz
I	Anzahl der Zwischenschichtneuronen
J	Anzahl der Resonanzlinien in dem 1D-Spektrum eines Spinsystems
L	Anzahl der Eingabeneuronen \simeq Gedächtnislänge eines Spinsystems
M	Anzahl der Trainingsmuster
c_{il}	Gewichtskoeffizient eines Neurons in der Zwischenschicht
v_i	Gewichtskoeffizient eines Neurons in der Ausgabeschicht
$\mathfrak{F}(z)$	Transferfunktion eines neuronalen Netzes in der Zwischenschicht
\tilde{y}_n	Approximierte Systemantwort des neuronalen Netzes
$\tilde{\Phi}[x(t)]$	Approximiertes Systemübertragungsfunktional des neuronalen Netzes
$\Im[z], \Re[z]$	Imaginärteil bzw. Realteil von z
$Y(\omega)$	Fouriertransformierte von $y(t)$
$\mathcal{Y}(\omega)$	Aus $Y(\omega)$ gefiltertes Spektrum
$\mathcal{Y}(t)$	Invers Fouriertransformierte von $\mathcal{Y}(\omega)$
CC	Kaskadenkorrelation (<i>cascade-correlation</i>)
$E.COSY$	Exklusive korrelierte Spektroskopie (<i>exclusive correlated spectroscopy</i>)
FID	Freier Induktionsabfall (<i>free induction decay</i>)
$RPROP$	Elastische Fortpflanzung (<i>resilient propagation</i>)
SER	Zeitdatensatz aus mehreren $FIDs$ einer $COSY$ -Messung
SW	Spektrale Breite
$TPPI$	Zeitproportionale Inkrementierung der Impulsphasen (<i>time-proportional phase incrementation</i>)

1 Einleitung

Die Methodiken und Anwendungsgebiete der NMR-Spektroskopie haben sich seit etwa 30 Jahren durch die technischen Fortschritte enorm vervielfältigt. Neben der Verbesserung der Magnete, Hf-Sender und Empfänger ist dies vor allem der Leistungssteigerung der Rechner zuzuschreiben (Speichergröße, Wortlängen, Rechenzeit). In der Praxis der NMR-Spektroskopie fand letzteres im wesentlichen Niederschlag in der Rechnersteuerung der konventionellen Mehrimpulsexperimente, der Berechnung der Spektren aus den gemessenen Zeitfunktionen und der sich anschließenden Datenanalyse. Die voraussehbar fortwährende Steigerung des Preis-Leistungs-Verhältnisses des Rechners führte allerdings früh dazu, neue NMR-methodische Ansätze zu entwickeln, die mit einem Weniger an experimentellem Aufwand (Einstellkomplexität, Messzeit, Fehlertoleranz) ein Mehr an spektraler Information zu Lasten eines erhöhten mathematischen Aufwands liefern. Als Beispiele seien hier die lineare Prädiktion [1, 2], die Bayes-Analyse [3, 4, 5] und der Einsatz der nichtlinearen Systemanalyse [6, 7, 8, 9, 10, 11] genannt. Sie alle lassen sich der *Chemometrie* zuordnen, einer noch jungen Disziplin der Chemie [12]. In der vorliegenden Arbeit wird dieser Weg weiter beschritten in Richtung der ^1H -NMR-Strukturaufklärung in flüssigen Systemen.

Ein Grundschrift der ^1H -NMR-Strukturaufklärung ist die Bestimmung der *Konnektivitäten* zweier Resonanzlinien eines eindimensionalen (1D) Spektrums bei den Frequenzen $\omega_1 = \Omega_m$ und $\omega_2 = \Omega_n$, deren Werte von den chemischen Verschiebungen δ_i und δ_k zweier H-Atome i und k des Moleküls und ihrer skalaren Spin-Spin-Koppelung J_{ik} bestimmt sind. Diese Konnektivitäten sind in einem 2D-H-Spektrum in Form von Kreuzsignalen an der Stelle $\omega_1 = \Omega_m$ und $\omega_2 = \Omega_n$ erkennbar und zeigen Teilstrukturen des Moleküls direkt an. Standardmessverfahren ist das konventionelle Doppel-Hf-Impulsexperiment [13, 14].

Charakteristisch für die konventionelle 2D-NMR ist, dass die Strukturanalyse erst mit dem 2D-Spektrum nach Maßgabe der erkennbaren Kreuzsignale durchführbar ist. Für vorab zu spezifizierende Aussagesicherheit ist eine feste Messzeit T erforderlich. Dies bedingt eine relativ lange Verweilzeit der Messprobe im kostspieligen supraleitenden Magneten. Dieser Nachteil besteht nicht in der 2D-NMR mit Rauschanregung [6, 7, 8], die Messung kann im Prinzip jederzeit beendet werden. Bestehen Zweifel, ob im resultierenden 2D-Spektrum eine Erhebung ein Kreuzsignal ist oder 2D-Rauschen vorliegt, werden weitere Messpunkte aufgenommen und zusammen mit den bestehenden analysiert. Dieser Vorteil konnte allerdings wegen der zeitaufwendigen Berechnung des 2D-Spektrums in der Praxis noch nicht realisiert werden. Beiden Verfahren ist überdies gemein, dass die Konnektivitäten erst durch Spektreninspektion feststellbar sind, sei es visuell oder per Mustererkennungsprogramm.

Grundlage der Untersuchungen dieser Dissertation ist die nichtlineare Systemanalyse, in der das Übertragungsfunktional durch ein neuronales Netz (*NN*) approximiert wird [9, 10, 11]. Dieser

neuartige mehrdimensionale Ansatz in der NMR-Signalanalyse, im weiteren mit *NN*-NMR angesprochen, steht etwa in der Mitte zwischen der konventionellen 2D-NMR und der 2D-NMR mit Rauschanregung. Die *NN*-NMR behält die Vorteile der 2D-NMR mit Rauschanregung (u.a. die Einfachheit und Fehlertoleranz bei der Hf-Anregung), vermeidet ihren Nachteil der langen Rechenzeiten und erscheint geeignet, die Probenverweilzeit substantiell zu verkürzen und hohe Empfindlichkeiten zu erreichen. Weiterhin hat sie bei passender Wahl der *NN*-Architektur das Potenzial für eine schnelle vollautomatische Strukturaufklärung: Die gesuchten Konnektivitäten werden hier direkt durch Größe und Vorzeichen der mit Hilfe der Messdaten ermittelten *NN*-Gewichte angezeigt, die Suche nach Kreuzsignalen im 2D-Spektrum ist nicht erforderlich. Dieses würde sich für Strukturaufklärungen, die nach dem Ausschlussprinzip durchgeführt werden (z.B. in der kombinatorischen Chemie) als nützlich erweisen. Offen ist allerdings noch, inwieweit dieses Potenzial tatsächlich in der Praxis anwendbar ist.

Der zeitbestimmende Schritt der *NN*-NMR ist die Ermittlung – auch Lernphase oder Training genannt – der reellen oder komplexen *NN*-Gewichte durch einen Trainingsalgorithmus. Während des Trainings, das am zweckmäßigsten in einem mit der NMR-Apparatur vernetzten Rechnercluster durchgeführt wird, werden alle das neuronale Netz charakterisierenden Gewichtskoeffizienten solange iterativ verändert, bis das Übertragungsfunktional des Spinsystems bestmöglich approximiert wird.

In der vorliegenden Dissertation wird die Methodik der *NN*-NMR weiterentwickelt und um neuartige Ansätze ergänzt. Ein wichtiger Schritt ist die Verkürzung der Rechenzeit für das Netztraining bei Vergrößerung der untersuchten Spinsysteme. Alternative, der Literatur entnommene Trainingsalgorithmen werden auf ihre Eignung überprüft. Ein weiterer Schwerpunkt ist die Nutzung von vorhandenen spektralen und chemischen Informationen, um die *NN*-NMR auch für die Überlagerung von verschiedenen Spinsystemen sowie Spinsystemen höherer Ordnung anwenden zu können. Aus dieser Verflechtung ergeben sich interessante Ansätze für eine vollautomatische ^1H -NMR-Strukturanalyse, die in dieser Arbeit vorgestellt werden.

2 Theoretische Grundlagen

2.1 Theorie der mehrdimensionalen NMR-Spektroskopie

2.1.1 Quantenmechanische Beschreibung

Ein Ensemble von Spinsystemen wird quantenmechanisch durch die reduzierte Dichtematrix $\sigma(t)$, die das Ensemblemittel der Zustände aller Einzelspinsysteme wiedergibt, und deren Zeitentwicklung, der Liouville-von-Neumann-Gleichung beschrieben [13]

$$\frac{\partial \sigma(t)}{\partial t} = i [\sigma(t), \mathbf{H}(t)] . \quad (2.1)$$

Der Hamiltonoperator $\mathbf{H}(t)$ (in Vielfachen von \hbar), setzt sich aus dem ungestörten Hamiltonoperator \mathbf{H}_0 und einem zeitabhängigen Störoperator $\mathbf{H}_1(t)$ zusammen

$$\mathbf{H}(t) = \mathbf{H}_0 + \mathbf{H}_1(t) . \quad (2.2)$$

In einem konstanten Magnetfeld besteht der zeitunabhängige Hamiltonoperator \mathbf{H}_0 ohne Berücksichtigung des chemischen Austausches und der Relaxation aus zwei Anteilen. Der Zeeman-Operator \mathbf{H}_Z beschreibt die Wechselwirkung der N Kerne l (Larmorfrequenz $\nu_l = 2\pi\Omega_l$) mit dem äußeren Feld, \mathbf{H}_J trägt der skalaren Spin-Spin-Koppelung zwischen den Kernen l und k Rechnung (Koppelungskonstante J_{lk})

$$\mathbf{H}_0 = \mathbf{H}_Z + \mathbf{H}_J \quad (2.3)$$

$$= \sum_{l=1}^N \nu_l \mathbf{I}_{lz} + \sum_{l=2}^N \sum_{k=1}^{l-1} J_{lk} \vec{\mathbf{I}}_l \vec{\mathbf{I}}_k . \quad (2.4)$$

Bei Einstrahlung eines Hochfrequenzimpulses (Hf-Impuls) der Magnetflussdichte \vec{B}_1 entlang der x-Achse des rotierenden Koordinatensystems lautet der Störterm \mathbf{H}_1

$$\mathbf{H}_1 = \sum_{l=1}^N \omega_1 \mathbf{I}_{lx} \quad (2.5)$$

$$\text{mit } \omega_1 = \gamma |\vec{B}_1| .$$

Die Kreisfrequenz ω_1 gibt die Amplitude des eingestrahlten Hf-Feldes an. Bei einer Impulsdauer τ wird die Gesamtmagnetisierung der Spins um den Kippwinkel $\alpha = \omega_1 \tau$ ($\omega_1 \gg \Delta\omega_{\text{offset}}$) gedreht. Bei Einstrahlung eines Impulses dominiert der Störterm \mathbf{H}_1 das Zeitverhalten des Spinsystems. Es gilt in guter Näherung $\mathbf{H}(t) = \mathbf{H}_1$. Vor und nach Impulsbestrahlung gilt

$$\mathbf{H}(t) = \mathbf{H}_0.$$

Ausgehend vom Dichteoperator σ_{eq} des thermischen Gleichgewichts erzeugt ein Impuls Kohärenzen, d.h. von Null verschiedene Nichtdiagonalelemente, von denen in der NMR-Spektroskopie nur die Einquantenkohärenzen direkt detektierbar sind. Dieses Signal $s_a(t)$ (mit $a = x, y, +, -$) wird durch den Erwartungswert von \mathbf{I}_a gemäß

$$s_a(t) = \langle \mathbf{I}_a \rangle \quad (2.6)$$

$$= \text{Tr}\{\mathbf{I}_a \sigma(t)\} \quad (2.7)$$

$$\text{mit } \mathbf{I}_+ = \mathbf{I}_x + i\mathbf{I}_y \quad \text{und} \quad \mathbf{I}_- = \mathbf{I}_x - i\mathbf{I}_y$$

beschrieben.

Die Rückkehr des Dichteoperators $\sigma(t)$ in den thermodynamischen Gleichgewichtszustand σ_{eq} wird dargestellt mit Hilfe des Relaxationssuperoperators Γ und dem Austauschsuperoperator Ξ , der die Einstellung eines dynamischen Gleichgewichts basierend auf chemischem Austausch beschreibt. Gleichung (2.1) erweitert sich damit zu

$$\frac{\partial \sigma(t)}{\partial t} = i[\sigma(t), \mathbf{H}(t)] + \{\Gamma + \Xi\}(\sigma(t) - \sigma_{eq}). \quad (2.8)$$

2.1.2 Eindimensionale NMR-Spektroskopie

In der NMR-Strukturaufklärung werden vor allem Kerne mit einer Kernspinqantenzahl $I = 1/2$ vermessen, zu denen z.B. ^1H , ^{13}C , ^{15}N , ^{19}F und ^{31}P gehören. In der eindimensionalen (1D) NMR-Spektroskopie werden die Kerne einer Probe durch einen Hf-Impuls aus ihrem thermodynamischen Gleichgewicht gebracht. Die resultierende Quermagnetisierung (Gleichung (2.7)) gibt Anlass zum Detektionssignal, dem *free induction decay (FID)*, das digitalisiert (Abtastintervall Δt) und auf Speichermedien gesichert wird. Der Signalabfall durch Relaxation und chemischen Austausch wird durch Gleichung (2.8) beschrieben. Abbildung (2.1) zeigt schematisch das eindimensionale Impulsexperiment.

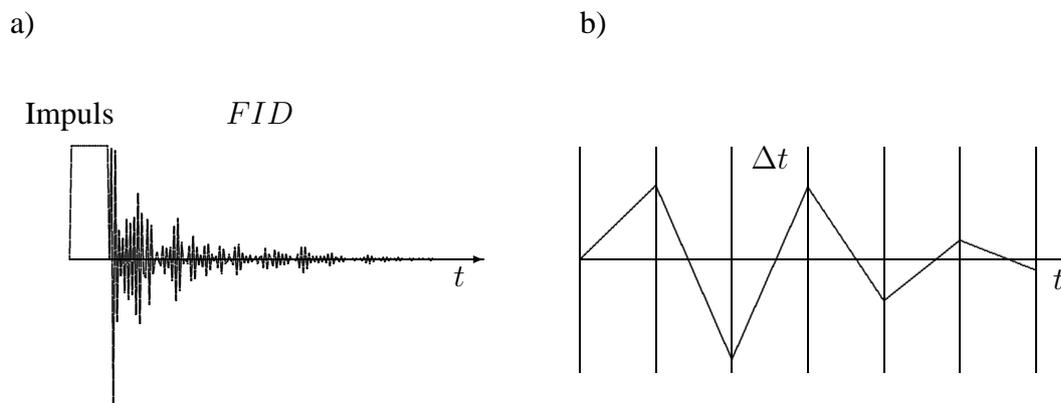


Abb. 2.1: Aufnahme eines Zeitsignals in einem 1D-Experiment: a) *FID* nach Hf-Impuls, b) Ausschnitt der ersten sechs Datenpunkte des *FID*s.

Die Fouriertransformation des *FIDs* liefert das Spektrum, aus dem die Resonanzfrequenzen der Einquantenkohärenzen des Spinsystems bestimmt werden können. Die Lage der Signale, ihre Feinstruktur, ihre Signalbreite und ihre Intensität lassen Rückschlüsse auf die Molekülstruktur und -dynamik zu.

Bei größeren Spinsystemen treten Überlagerungen der Signale auf, die eine Analyse erschweren, wenn nicht sogar unmöglich machen. Abhilfe schaffen mehrdimensionale Experimente, bei denen Zeit- und Messaufwand jedoch größer sind.

2.1.3 Mehrdimensionale NMR-Spektroskopie

Zweidimensionale NMR-Spektroskopie

Ein zweidimensionales (2D) NMR-Experiment umfasst fünf Phasen [17]:

1. Wartezeit (kein Hf-Impuls),
2. Präparation (Hf-Impulse, Verzögerungszeiten),
3. Evolution (kein Hf-Impuls, keine Datenaufnahme, $t_1 = m_1 \Delta t$ mit $m_1 = 1, \dots, M_1$),
4. Mischung (Hf-Impulse, Verzögerungszeiten),
5. Detektion (Datenaufnahme, $t_2 = m_2 \Delta t$ mit $m_2 = 1, \dots, M_2$).

In der Wartezeit stellt sich das thermische Gleichgewicht ein. In der Präparations- und Mischungsphase können eine Vielzahl von Hf-Impulsen auf das Spinsystem einwirken. Dies führt zu einer großen Anzahl unterschiedlicher 2D-Experimente [18, 19].

Die fünf Phasen werden M_1 mal wiederholt, wobei die Zeit t_1 während der Evolution sukzessiv um Δt erhöht wird. In der Detektionszeit t_2 werden M_1 *FIDs* mit jeweils M_2 reellen oder komplexen Datenpunkten aufgenommen (Einphasen- bzw. Quadratur-Nachweis). Die resultierende ($M_1 \times M_2$)-Datenmatrix kann zur Auswertung mit Hilfe einer zweidimensionalen Fouriertransformation bezüglich t_1 und t_2 in ein 2D-Spektrum überführt werden. Aufgrund von Kohärenztransfers durch die Hf-Impulse in der Präparations- und Mischungsphase kommt es zu einer Aufspaltung überlagerter Signale in eine zweite Frequenzachse.

Correlated Spectroscopy (COSY)

Bei dem einfachsten zweidimensionalen NMR-Experiment bestehen sowohl die Präparations- als auch die Mischungsphase aus je einem Hf-Impuls mit dem Kippwinkel α und der Phase φ [20]

$$\text{Wartezeit} - \alpha_\varphi - t_1 - \alpha_\varphi - t_2 \text{ (FID)}.$$

In Abbildung (2.2) ist dieses Experiment schematisch dargestellt. Dabei ist jeweils ein *FID* in t_1 und t_2 eingezeichnet, detektiert wird lediglich in t_2 . Eine Detektion während der Evolutionsphase t_1 wäre denkbar, ist aber wegen eingeschränkter Verwendbarkeit in der Praxis unüblich.

Mit Hilfe des Produktoperatorformalismus können im Fall zweier koppelnder Kerne A und X mit der Koppelungskonstante J_{AX} die zu messenden Größen einfach beschrieben werden [21].

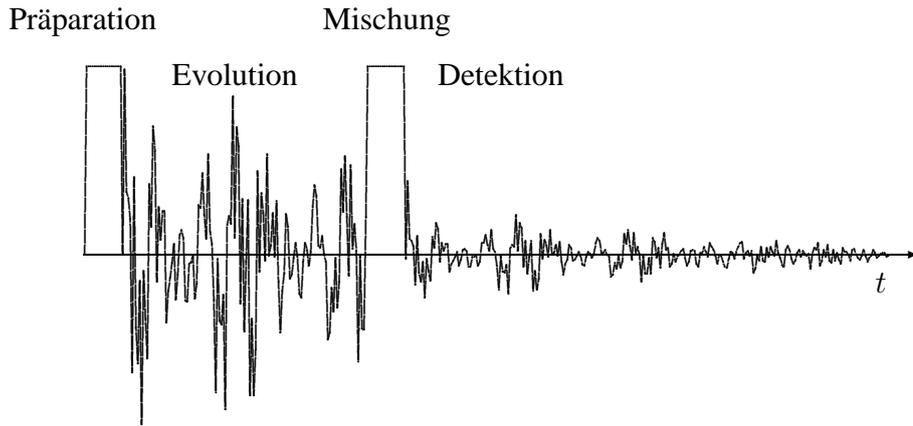


Abb. 2.2: Aufnahme eines *COSY*-Experiments.

Mit dem Kippwinkel $\alpha = 90^\circ$ für beide Hf-Impulse in Präparations- und Mischungsphase und unter Vernachlässigung der Relaxation lauten die detektierbaren Signale

$$S_D(t_1, t_2) = \sin(\Omega_A t_1) \cos(\pi J_{AX} t_1) \sin(\Omega_A t_2) \cos(\pi J_{AX} t_2) \quad (2.9)$$

$$S_K(t_1, t_2) = \sin(\Omega_X t_1) \cos(\pi J_{AX} t_1) \sin(\Omega_A t_2) \cos(\pi J_{AX} t_2) . \quad (2.10)$$

Gleichung (2.9) führt zu den Diagonalsignalen bei den Frequenzen $\omega_1 = \omega_2 = \Omega_A \pm \pi J_{AX}$. Sie entsprechen den Signalen im eindimensionalen Spektrum. Gleichung (2.10) entspricht den Kreuzsignalen $\omega_1 = \Omega_X \pm \pi J_{AX}$ und $\omega_2 = \Omega_A \pm \pi J_{AX}$, die die Spinkonnektivität anzeigen.

Werden für die Hf-Impulse Kippwinkel $\alpha < 90^\circ$ gewählt, so kann eine verkürzte Messung durchgeführt werden, da die Kerne schneller wieder in ihr thermisches Gleichgewicht gelangen. Zusätzlich erscheinen unerwünschte Axialsignale bei $\omega_1 = 0$, eine Folge der T_1 -Relaxation während der Evolutionszeit. Ein einfacher Phasenzklus, bestehend aus zwei Experimenten (FID_I und FID_{II}) mit verschiedenen Hf-Impulsen gleichen Kippwinkels α und den Phasen 0° und 180° während jeder Mischungsphase, und Summenbildung $FID_I + FID_{II}$ unterdrückt diese störenden Effekte.

$$\begin{aligned} & \text{Wartezeit} - \alpha_{0^\circ} - t_1 - \alpha_{0^\circ} - t_2 \text{ (} FID_I \text{)} \\ & \text{Wartezeit} - \alpha_{0^\circ} - t_1 - \alpha_{180^\circ} - t_2 \text{ (} FID_{II} \text{)}. \end{aligned}$$

Semisoft-COSY

Bei dieser Variante werden in der Präparationsphase mit selektiven Impulsen eine Auswahl an Kernen angeregt [22]. Während der Mischungsphase wird das gesamte Spinsystem erfasst. Das resultierende Spektrum zeigt nur den selektierten Bereich in ω_1 und den gesamten Bereich in ω_2 . Um weitere Bereiche zu selektieren, müssen erneute Messungen erfolgen.

COSY mit Quadratur-Detektion

Wird in der 1D-NMR die Quermagnetisierung in der x - und y -Achse des rotierenden Koordinatensystems detektiert, so kann gegenüber der Einphasendetektion eine Vorzeichenunterscheidung der gemessenen Frequenzen erfolgen, sodass die Trägerfrequenz in der Mitte des Spektrums positioniert werden kann. Für ein Einspinsystem können die beiden detektierten Signale $s_x(t)$ und $s_y(t)$ gemäß

$$s_x(t) = \cos(\Omega_A t) e^{-\frac{t}{T_2}} \quad (2.11)$$

$$s_y(t) = \sin(\Omega_A t) e^{-\frac{t}{T_2}} \quad (2.12)$$

$$s_{\text{Quad}}(t) = s_x(t) + i s_y(t) = e^{i\Omega_A t} e^{-\frac{t}{T_2}} \quad (2.13)$$

zu einem komplexen Signal $s_{\text{Quad}}(t)$ zusammengefasst werden. Eine komplexe Fouriertransformation liefert nur das Signal bei $\omega = +\Omega_A$.

In der 2D-Spektroskopie ist die Quadratur nur in der Beobachtungszeit t_2 möglich. Für eine Messung in t_1 -Quadratur sind zwei Verfahren gebräuchlich. Bei dem *time-proportional phase incrementation (TPPI)* wird die Phase des ersten Impulses nach jeder t_1 -Inkrementierung um 90° gedreht [23, 24]. Die Phase des zweiten Impulses bleibt konstant. Diese Impulsfolge führt zu einem alternierenden Sinus- und Kosinuscharakter der korrespondierenden *FIDs*. Eine reelle Fouriertransformation in t_1 ergibt ein amplitudenmoduliertes Signal.

Für das zweite Verfahren sind zwei Experimente erforderlich [25], dessen Signale aber getrennt gespeichert werden müssen

$$\begin{aligned} & \text{Wartezeit} - \alpha_{0^\circ} - t_1 - \alpha_{0^\circ} - t_2 \text{ (SER}_I\text{)} \\ & \text{Wartezeit} - \alpha_{90^\circ} - t_1 - \alpha_{90^\circ} - t_2 \text{ (SER}_{II}\text{)}. \end{aligned}$$

Die *FIDs* im Datensatz SER_I sind mit $\cos \Omega t_1$ moduliert, im SER_{II} mit $\sin \Omega t_1$. Um phasenverdrehen Signale zu vermeiden, müssen die beiden *SER*-Datensätze getrennt komplex fouriertransformiert werden. Die beiden Realteile werden zu einem neuen komplexen Signal zusammengesetzt und erneut komplex fouriertransformiert. Das auf diese Weise berechnete 2D-Spektrum enthält nur noch phasensensitive Signale in Absorptionsform.

2.2 Systemtheorie

2.2.1 Charakterisierung

In der Systemtheorie wird ein System durch ein Übertragungsfunktional $\Phi[x(t)]$ beschrieben, welches eine zeitabhängige Anregungsfunktion $x(t)$ auf eine Antwortfunktion $y(t)$ abbildet (siehe auch Abbildung (2.3))

$$y(t) = \Phi[x(t)]. \quad (2.14)$$

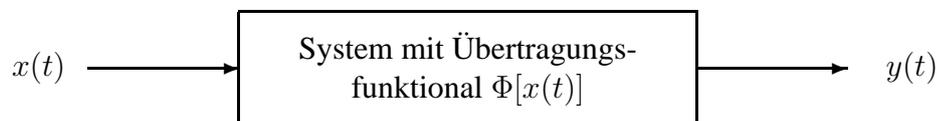


Abb. 2.3: Abbildung einer Anregung $x(t)$ auf eine Antwort $y(t)$.

Ein System ist zeitinvariant, wenn eine um ein Zeitinkrement Δt verschobene Anregung $x(t + \Delta t)$ eine um das gleiche Inkrement verschobene Antwortfunktion $y(t + \Delta t)$ liefert. Ein System ist kausal, wenn die Systemantwort nur von der Anregung zum momentanen oder früheren Zeitpunkt abhängt. Ein stabiles System liegt vor, wenn die Systemantwort nicht divergiert.

Die analytische Form des Funktionals in Gleichung (2.14) ist i. Allg. unbekannt. Es existieren verschiedene Modelle, die gemäß gemessener Systemantwort $y(t)$ und definierter Anregung $x(t)$ quantifiziert werden können. Ein mögliches Modell versucht, die im quadratischen Mittel beste Approximation an das Übertragungsfunktional $\Phi[x(t)]$ zu bekommen.

Bei Verwendung von gleichen diskreten Zeitintervallen Δt kann eine kontinuierliche Zeitfunktion $x(t)$ durch diskrete Werte $x(0), x(1), \dots, x(L-1)$ der Länge $L\Delta t$ beschrieben werden.

2.2.2 Lineare Systeme

Ein System, für welches das Superpositionsprinzip gilt, wird als linear bezeichnet

$$\Phi_{\text{lin}} [ax_1(t) + bx_2(t)] = a\Phi_{\text{lin}} [x_1(t)] + b\Phi_{\text{lin}} [x_2(t)] . \quad (2.15)$$

Besitzt das System ein endliches Gedächtnis der Länge $L\Delta t$, so kann eine einzelne Systemantwort $y_{\text{lin}}(m)$ aus den diskreten Anregungswerten $x(m)$ bis $x(m-L+1)$ abgebildet werden

$$y_{\text{lin}}(m) = \Phi_{\text{lin}} = \sum_{l=0}^{L-1} b_l x(m-l) = b_0 x(m) + b_1 x(m-1) + \dots + b_{L-1} x(m-L+1) . \quad (2.16)$$

Liegt die gesamte Systemantwort $\vec{y}_{\text{lin}} = (y_{\text{lin}}(M), y_{\text{lin}}(M-1), \dots)$ vor, so kann Gleichung (2.16) in Matrixschreibweise formuliert werden

$$\vec{y}_{\text{lin}} = \mathbf{X}\vec{b} . \quad (2.17)$$

Das untersuchte System wird durch den Vektor \vec{b} charakterisiert. Er beinhaltet die gleichen Koeffizienten wie in Gleichung (2.16). Für $M \geq L$ kann \vec{b} bestimmt werden.

2.2.3 Nichtlineare Systeme

Bei einem nichtlinearen System kann die Antwort $y(t)$ auf eine Anregung $x(t)$ durch einen Satz von Basisfunktionalen in eine Funktionalreihe der Form

$$F[x(\tau)] = \sum_{n=0}^{\infty} F_n[x(\tau)] \quad (2.18)$$

entwickelt werden. Ein geeigneter Satz von Basisfunktionalen F_n kann durch Faltungsintegrale zwischen nD -Zeitfunktionen $k_n(\tau_1, \dots, \tau_n)$ und der Anregungsfunktion $x(t)$ geschrieben werden [26]. Diese speziellen Basisfunktionalen sind homogen vom Grad n in $x(t)$ und werden als Volterra-Funktionalen V_n bezeichnet

$$V_n [k_n(\tau_1, \dots, \tau_n); x(\tau)] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} k_n(\tau_1, \tau_2, \dots, \tau_n) \times \\ x(t - \tau_1) x(t - \tau_2) \dots x(t - \tau_n) d\tau_n \dots d\tau_2 d\tau_1 . \quad (2.19)$$

Bei kausalen Systemen sind die nD Zeitfunktionen $k_n(\tau_1, \dots, \tau_n)$, die so genannten Volterra-Kerne, für negative Zeitargumente τ_i gleich Null. Die aus Gleichung (2.18) und (2.19) resultierende Volterra-Reihe

$$y(t) = \sum_{n=0}^{\infty} \int_0^{\infty} \int_0^{\infty} \dots \int_0^{\infty} k_n(\tau_1, \tau_2, \dots, \tau_n) \times \\ x(t - \tau_1) x(t - \tau_2) \dots x(t - \tau_n) d\tau_n \dots d\tau_2 d\tau_1 \quad (2.20)$$

beschreibt über die Gesamtheit der Volterra-Kerne das Verhalten des Systems. Die Volterra-Kerne sind symmetrisch bezüglich der Vertauschbarkeit der Zeitargumente, so ist z.B.

$$k_2(\tau_1, \tau_2) = k_2(\tau_2, \tau_1) . \quad (2.21)$$

Es ist bisher keine Vorschrift bekannt, um aus einer NMR-Messungen die Volterra-Kerne für ein gegebenes Spinsystem zu bestimmen.

2.2.4 NMR-Systeme

Ein Spinsystem ist ein nichtlineares, stabiles System, das zeitinvariant und kausal ist. Sein Übertragungsfunktional ist eine ungerade Funktion. Daher verschwinden alle Terme, die einen Volterra-Kern k_{2n} gerader Ordnung beinhalten [6]. Die Funktionalreihe (2.20) reduziert sich zu

$$\begin{aligned} y(t) &= \int_0^{\infty} k_1(\tau)x(t - \tau) d\tau \\ &+ \int_0^{\infty} k_3(\tau_1, \tau_2, \tau_3)x(t - \tau_1)x(t - \tau_2)x(t - \tau_3) d\tau_1 d\tau_2 d\tau_3 \\ &+ \dots \end{aligned} \quad (2.22)$$

Unter der Annahme, dass das Spinsystem ein Gedächtnis der Länge $L\Delta t$ besitzt, kann die Gleichung (2.22) in die diskrete Schreibweise

$$\begin{aligned} y(m) &= \sum_{l=0}^{L-1} k_1(l)x(m - l) \\ &+ \sum_{l_1=0}^{L-1} \sum_{l_2=0}^{L-1} \sum_{l_3=0}^{L-1} k_3(l_1, l_2, l_3)x(m - l_1)x(m - l_2)x(m - l_3) \\ &+ \dots \end{aligned} \quad (2.23)$$

überführt werden.

Der Kern $k_1(t)$ enthält die gleiche Information wie ein herkömmlich gemessenes *FID*-Signal [27, 28]. Entsprechend ist die Fouriertransformierte $K_1(\omega)$ von $k_1(t)$ ein dem gemessenen 1D-NMR-Spektrum äquivalent. Im dreidimensionalen Spektrum $K_3(\omega_1, \omega_2, \omega_3)$, welches aus $k_3(\tau_1, \tau_2, \tau_3)$ durch Fouriertransformation hervorgeht, sind nur Signale aus verbundenen Übergängen enthalten [7].

Die Diagonalebene

$$E_2(\omega_1, \omega_2) = K_3(\omega_1, \omega_2, -\omega_2) \quad (2.24)$$

enthält aufgrund von Spin-Spin-Koppelungen Signale, welche denen eines *E.COSY*-Spektrums entsprechen [29, 30] (siehe auch Abbildung (2.4a)).

Das *E.COSY* unterdrückt nicht verbundene Übergänge. Der Parallelschnitt

$$\begin{aligned} P_2(\omega_2, \omega_3) &= K_3(\omega_r, \omega_2, \omega_3) \\ \text{mit } \omega_r &= \text{Frequenz eines detektierbaren Einquantenübergangs} \end{aligned} \quad (2.25)$$

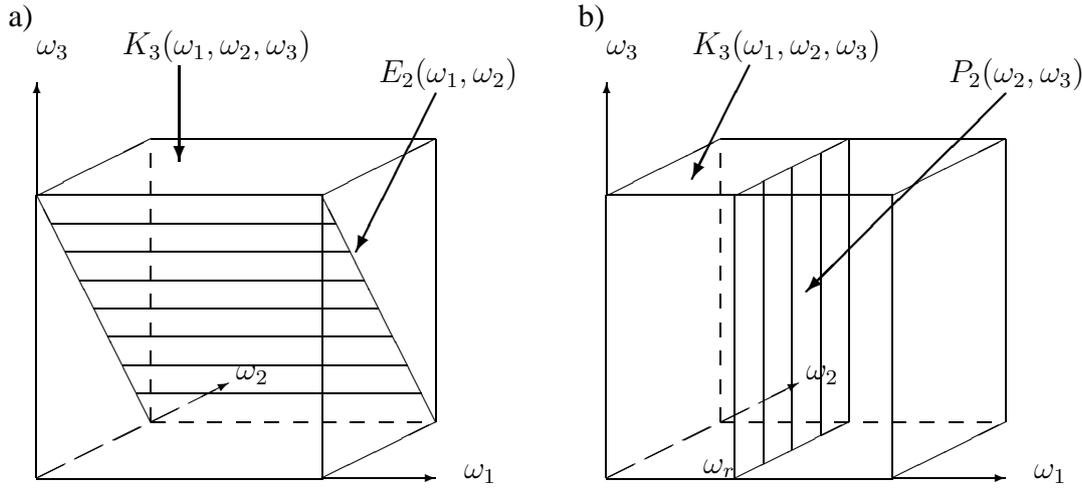


Abb. 2.4: a) Diagonalebene $E_2(\omega_1, \omega_2)$ und b) Parallelschnitt $P_2(\omega_2, \omega_3)$ des Kerns $K_3(\omega_1, \omega_2, \omega_3)$.

in Abbildung (2.4b) liefert zusätzlich Konnektivitäten zu den von dort aus direkt erreichbaren Zuständen [31].

Näherungsansatz für einen Spin mit $I = 1/2$

Im Fall des einfachsten Spinsystems kann die detektierbare Magnetisierung $M^+(t)$ auf Grundlage der Blochschen Gleichungen als Summe über Produkte zweier Basisfunktionale angenähert werden [32]

$$M^+(t) = iM_0 (P - PQP + PQPQP - \dots) \quad (2.26)$$

$$\text{mit } P = P(t) = \int_{-\infty}^{\infty} e^{-\left(\frac{1}{T_2} + i\Omega\right)\tau} \omega_1(t - \tau) d\tau \quad (2.27)$$

$$Q = Q(t) = \int_{-\infty}^{\infty} e^{-\frac{T_2}{T_1} \left(\frac{1}{T_2} - i\Omega\right)\tau} \omega_1(t - \tau) d\tau . \quad (2.28)$$

Dieser Ansatz beschreibt das Sättigungsverhalten eines einzelnen Kerns mit der komplexen Resonanzfrequenz Ω und der Anregungsfunktion $\omega_1(t)$. Die unendliche Summe in Gleichung (2.26) kann für $P(t) < 1$ und $Q(t) < 1$ in einen geschlossenen Ausdruck der Form

$$M^+(t) = iM_0 \frac{P(t)}{1 + P(t)Q(t)} \quad (2.29)$$

überführt werden. Für $T_1 \approx T_2$, was in guter Näherung für viele flüssige Proben angenommen werden kann, vereinfacht sich Gleichung (2.29) zu

$$M^+(t) = iM_0 \frac{P(t)}{1 + |P(t)|^2} . \quad (2.30)$$

Das durch Gleichung (2.30) beschriebene System ist eine Kaskade aus einem linearen System mit Gedächtnis, gefolgt von einem nichtlinearen System ohne Gedächtnis (Abbildung (2.5)).

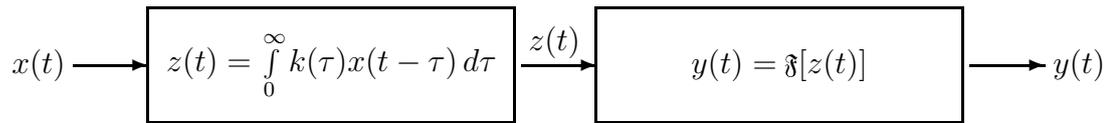


Abb. 2.5: Kaskade zweier Systeme zur Beschreibung von Spinsystemen ohne skalare Koppelung.

Näherungsansatz für ein Mehrspinsystem ohne Koppelung

Eine Parallelschaltung von I mit v_i gewichteten Kaskaden ist in der Lage, ein System von mehreren Spinsystemen ohne Koppelung zu beschreiben [11]

$$y(t) = \Phi[x(t)] = \sum_{i=1}^I v_i \mathfrak{F}[z_i(t)] \quad (2.31)$$

$$z_i(t) = \int_0^{\infty} k_i(\tau)x(t-\tau) d\tau. \quad (2.32)$$

Näherungsansatz für ein Mehrspinsystem mit Koppelung

Da koppelnde Spins voneinander abhängig sind, können sie nicht durch das Superpositionsprinzip dargestellt werden. Mit der Einführung zusätzlicher wechselwirkende Schaltungen c_{il} kann eine diskrete Schreibweise für ein Spinsystem mit der Gedächtnislänge $L\Delta t$

$$y_m = \sum_{i=1}^I v_i \mathfrak{F} \left[\sum_{l=1}^L c_{il} x_{m-l+1} \right] \quad (2.33)$$

erhalten werden. Diese Formulierung gleicht der mathematischen Beschreibung der so genannten künstlichen neuronalen Netze, ein von der Neurobiologie entlehntes Modellsystem [33,34]. Gleichung (2.33) kann als ein neuronales Netz mit einer verborgenen Schicht aufgefasst werden. Die nichtlineare Funktion $\mathfrak{F}[z_{i,m}]$ wird im Sprachgebrauch der neuronalen Netze als Transferfunktion bezeichnet, die Koeffizienten c_{il} und v_i als ihre Gewichte.

2.3 Neuronale Netze

2.3.1 Grundlagen

Ein Charakteristikum der Problemlösung in biologischen Systemen ist die Adaption. Durch Lernprozesse können Handlungsweisen zur Problemlösung formuliert werden. Adaptierende Systeme wie neuronale Netze sind der Natur nachempfundene Algorithmen.

Neuronale Netze haben sich aufgrund attraktiver Eigenschaften in den verschiedensten wissenschaftlich-technischen Bereichen etabliert. Dazu gehören neben ihrer Lernfähigkeit ihre Robustheit, Fehlertoleranz und Generalisierungsfähigkeit. Es gilt jedoch immer zu prüfen, ob für ein gegebenes Anwendungsproblem ausschließlich mit Netzen effektive Lösungen erzielt werden können. Bei manchen Fragestellungen bieten sich Kombinationen mit anderen Methoden (z.B. Expertensysteme, Fuzzy-Logik) an.

Eine Vielzahl von Problemlösungen durch neuronale Netze kann nicht kontrolliert werden, da Merkmale für ein erfolgreiches Training zunächst nicht bekannt sind. Eine Qualitätskontrolle der abgefragten Daten ist deswegen unabdingbar.

Die grundlegende Funktionsweise des neuronalen Netzes ist die des Klassifikators. Aus der Sicht des Anwenders ergeben sich verschiedene Problemklassen:

- Musterklassifikation,
- Kategorisierung (*Clustering*),
- Funktionsapproximation,
- Prognose,
- Optimierung,
- Inhaltsbasierte Speicherung,
- Steuerung und Regelung.

Zu jeder Problemklasse existieren Applikationen mit verschiedenen Netzarchitekturen [35].

In der Chemie haben neuronale Netze z.B. ihren praktischen Einsatz in der Prozessregelung, in der Reaktionsprognose und bei der Mustererkennung spektroskopischer Daten gefunden [36, 37].

2.3.2 Neuronale Netze in der NMR-Signalanalyse (NN-NMR)

Topologie der eingesetzten neuronalen Netze

Die in dieser Arbeit verwendeten neuronalen Netze gehören zur Klasse der funktionsapproximierenden Netze. Auf der Basis des NMR-Messfunktion ist auf bekannte Ein- und Ausgabetupel (x_m, y_m) ein unbekanntes Abbildungsfunktional $\Phi[x_m]$ zu erlernen. Das lernende Netz bildet eine approximierte Ausgabe \tilde{y}_m ab. Ein trainiertes Netz muss sowohl die bekannten Tupel reproduzieren, als auch neue bis dahin unbekannte Funktionswerte hinreichend genau approximieren können. Für die Analyse von Spinsystemen haben sich rückkoppelungsfreie *feedforward*-Netze erster Ordnung bewährt [10] (Abbildung (2.6)). Es bestehen nur gerichtete Verbindungen zwischen der ersten und den nachfolgenden Schichten.

Die Eingabeschicht repräsentiert L sequenzielle Anregungswerte $x_m, x_{m-1}, \dots, x_{m-L+1}$ entsprechend der Gedächtnislänge $L\Delta t$ des Spinsystems. Die Anregung erfolgt mit einer Sequenz von Hf-Impulsen. Alle Impulse besitzen den gleichen Kippwinkel zwischen 15° und 60° . Bei einer Einphasendetektion ohne Vorzeichenunterscheidung der Frequenzen betragen die Impulsphasen 0° und 180° und für die NMR-Messung muss die Trägerfrequenz am Rande des Spektrums positioniert werden.

Durch die Einführung von Impulsen mit den Phasen 90° und 270° ist eine Unterscheidung von positiven und negativen Frequenzen in höherer Ordnung möglich [11]. Um eine Quadraturdetektion durchzuführen, muss das neuronale Netz in der Eingabeschicht vergrößert werden.

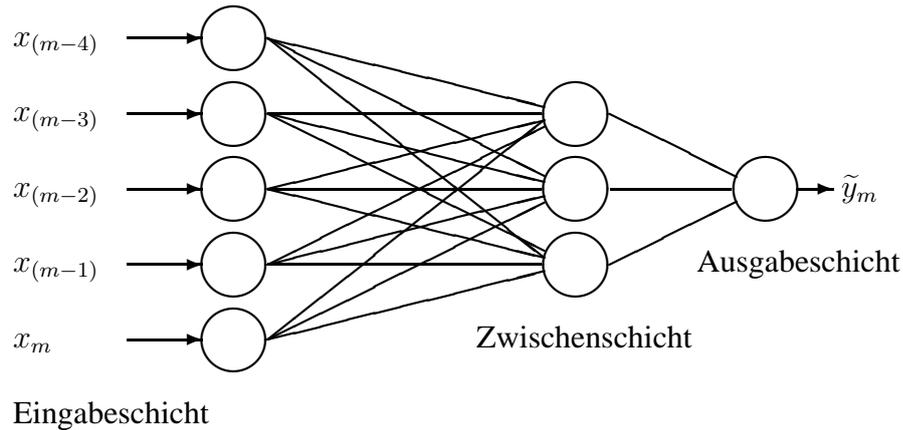


Abb. 2.6: Neuronales Netz mit *feedforward*-Topologie. Die Größe für die Eingabeschicht ist $L = 5$.

Dieses geschieht durch komplexe Eingabevektoren. Der Realteil beinhaltet Impulse der Phasen 0° und 180° , der Imaginärteil die der Phasen 90° und 270° . Anstelle der reellen Gewichte besitzt das neuronale Netz komplexe Gewichte in allen Schichten.

Die Ausgabeschicht besteht aus einem Neuron. Sein Ausgabewert \tilde{y}_m repräsentiert die Antwort y_m des Spinsystems – das NMR-Messsignal – zu den x_m, \dots, x_{m-L+1} Anregungswerten. Bei Einfach-Phasendetektion ist dies ein reeller Wert, bei Quadraturdetektion ein komplexer Wert.

Die Eingabewerte werden von den verdeckten Neuronen in der Zwischenschicht mit einem Gewicht c_{il} multipliziert und summiert. Eine Transferfunktion bildet die Summe auf die Ausgabeschicht ab. Diese multipliziert wieder die ankommenden Funktionswerte der einzelnen Neuronen mit ihrem Gewicht v_i und summiert sie zu einem Ausgabewert \tilde{y}_m (Abbildung (2.7)).

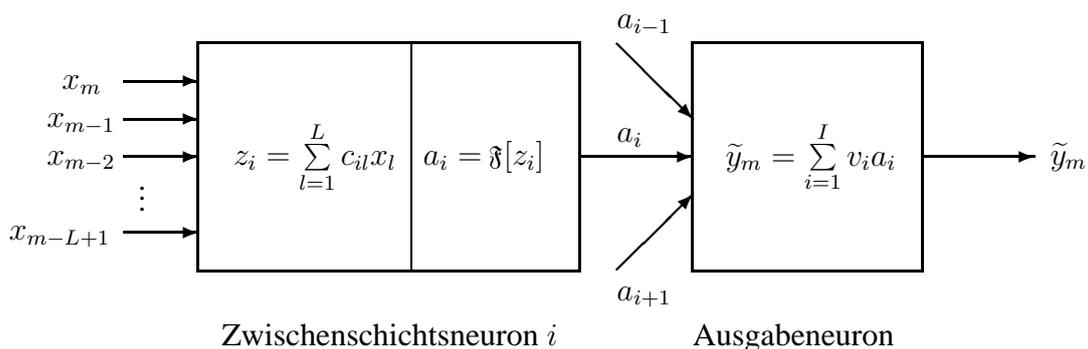


Abb. 2.7: Ausschnitt aus einem *feedforward*-Netz zur Funktionalapproximierung in der NMR-Signalanalyse.

Dieses Vorgehen entspricht der aus der Systemtheorie hergeleiteten Gleichung (2.33). Die bei allgemeinen Netzen üblichen Schwellenwerte (*bias*), die eine Werteaddition auf die gewichtete Eingabe vornehmen, sind bei NMR-Systemen gleich Null [10].

Wahl der Transferfunktion

Üblicherweise werden für neuronale Netze sigmoide Transferfunktionen

$$\mathfrak{f}(z) = \frac{1}{1 + e^{-\frac{z}{g}}} \quad g = \text{konst.} \quad (2.34)$$

$$\mathfrak{f}(z) = \tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} \quad (2.35)$$

zur Funktionsapproximierung benutzt. Neben ihrer Stetigkeit und Monotonie sind sie überall differenzierbar. Für die NMR besser geeignet sind allerdings die nicht-monotonen Funktionen [10]

$$\mathfrak{f}(z) = \frac{z}{1 + z^2} \quad (2.36)$$

$$\mathfrak{f}(z) = ze^{-z^2}. \quad (2.37)$$

Treten in einem Zeitintervall zu viele Anregungen auf, so wird das Spinsystem gesättigt und der Beitrag der Systemantwort nimmt entsprechend Gleichung (2.30) ab. Bei vorzeichensensitiver Detektion werden Real- und Imaginärteil separat transformiert. Das Ausgabeneuron besitzt in allen Netzarchitekturen dieser Arbeit die Transferfunktion $\mathfrak{f}(z) = z$.

Eine Untersuchung der Beiträge unterschiedlicher Ordnung der nichtlinearen Antwort des Spinsystems auf die Hochfrequenzanregung ergibt, dass das neuronale Netz nur Beiträge erster und dritter Ordnung zu erfassen hat [9]. Diese Separation zwischen erster und dritter Ordnung lässt sich durchführen, indem bei einem einschichtigen neuronalen Netz die linearen und die nichtlinearen Beiträge voneinander getrennt werden. Da gemäß Gleichung (2.15) das Superpositionsprinzip gilt, werden die linearen Beiträge durch ein Neuron mit der Transferfunktion $\mathfrak{f}(z) = z$ erfasst. Die Beiträge dritter Ordnung werden durch mehrere Neuronen mit der Transferfunktion $\mathfrak{f}(z) = z^3$ repräsentiert. Die Separation erfolgt bei Quadraturdetektion analog der obigen Transferfunktionen mit der Identität eines Neurons für den linearen Beitrag und mit $\mathfrak{f}(z) = zz^*z$ für die übrigen Neuronen [11].

Lernphase (Training)

Der numerische Zusammenhang muss zwischen Ein- und Ausgangsgrößen trainiert werden. Während dieser Lernphase werden die Gewichte in den Neuronen iterativ solange verändert, bis die Ausgangsgrößen \tilde{y}_m bestmöglich im Rahmen eines Konvergenzkriteriums der zu erlernenden Antwort y_m entspricht [38]. Dieses Lernverfahrens wurde als *backpropagation*-Algorithmus bekannt und trug zur weiten Verbreitung der neuronalen Netze bei [39]. Gewöhnlich werden die Gewichte nach jeder Präsentation einzelner Muster angepasst (*online*-Training).

Der Lernalgorithmus für die Signalanalyse eines Spinsystems mittels eines reellen neuronalen Netzes mit einer verdeckten Schicht und einem Ausgabeneuron lässt sich wie folgt beschreiben:

1. Initialisierung der beiden Gewichtsmatrizen (q Neuronen der vorhergehenden Schicht) mit Zufallszahlen im Intervall $[-\sqrt{\frac{1}{q}}, \sqrt{\frac{1}{q}}]$ [34].
2. Festlegen des Konvergenzkriteriums, Initialisierung der *backpropagation*-Lernparameter η und μ , Einlesen des gesamten Anrege- und Antwortvektors, Angabe der Gedächtnislänge L für das Spinsystem.

3. Beginn der Lernphase:

- a) Beginn des Lernzyklus.
- b) Zufällige Auswahl eines Musters m mit den Eingabewerten $x_m, x_{m-1}, \dots, x_{m-L+1}$ und dem dazugehörigen Antwortwert y_m .
- c) Vorwärtsaktivierung, Berechnung der I verdeckten Neuronenaktivitäten a_i und des approximierten Ausgabewerts \tilde{y}_m :

$$a_i = \mathfrak{f}(z_i) \text{ mit } z_i = \sum_{l=1}^L c_{il}x_l \text{ und } \tilde{y}_m = \sum_{i=0}^I v_i a_i .$$
- d) Lernfehlerberechnung der Ausgabe: $\mathcal{E}_m = y_m - \tilde{y}_m$.
- e) Fehlerreduktion um Instabilitäten durch zu große Fehler zu vermeiden (c ist ein geeignet gewählter Parameter):

$$E_m^{out} = \frac{\mathcal{E}_m}{1+c\mathcal{E}_m^2} .$$
- f) Gewichtsmodifikationen für die Zwischenschichtneuronen:

$$c_{il} = c_{il} + \Delta c_{il} \text{ mit } \Delta c_{il} = \eta \mathfrak{f}'(z_i) E_m^{out} v_i + \mu c_{il}^{\text{vorherigerZyklus}} .$$

 Durch die Konstante μ werden Oszillationen um ein Minimum in der Fehlerhyperfläche bei zu großer Lernrate η vermieden (Impulsterm)¹.
- g) Gewichtsmodifikationen für das Ausgabeneuron:

$$v_i = v_i + \Delta v_i \text{ mit } \Delta v_i = \eta a_i E_m^{out} + \mu v_i^{\text{vorherigerZyklus}} .$$
- h) Sind weitere mögliche Muster vorhanden, zurück zu Schritt (b), ansonsten ist der Lernzyklus abgeschlossen.
- i) Verkleinerung der Lernrate η um 10%, wenn die Fehlerreduktion stagniert.
- j) Wiederholung der Lernzyklen (a)-(i) bis Konvergenzkriterium erfüllt oder eine maximale Zyklenanzahl erreicht ist.

Die Lernphase für ein komplexes Netz verläuft in ähnlicher Weise [11].

2.4 Methodik der NN-NMR

Das Verfahren besteht aus drei Schritten (Abbildung (2.8)).

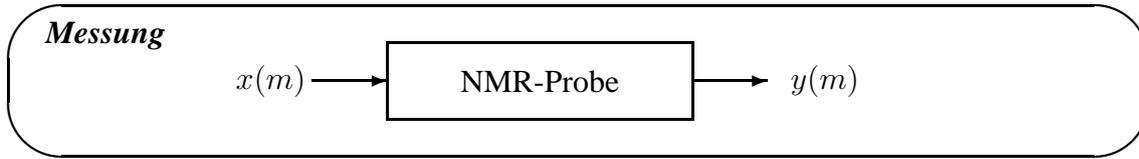
2.4.1 Messung der Spinsystemdaten

Die Probe wird im Spektrometer mit einer Sequenz von Hochfrequenzimpulsen angeregt. Zur Kodierung wird eine Zahlenfolge verwendet. Jeder Zahlenwert repräsentiert einen Wert der Anregung, der in Abständen von Δt über die Senderspule im Spektrometer auf die NMR-Probe eingestrahlt wird. Die reelle bzw. komplexe Antwort des Spinsystems wird in den gleichen Abständen Δt aufgezeichnet.

Bei Einfach-Phasendetektion wird eine ternäre zufällige Folge der Zahlen 0, +1 und -1 als Anregung verwendet. Bei 0 wird kein Impuls gesendet, +1 und -1 senden Impulse mit den

¹Die Einführung eines Impulsterm entspricht einer Verbesserung des ursprünglichen *backpropagation*-Algorithmus, der im weiteren Verlauf verkürzt als *backpropagation* bezeichnet wird.

Spektrometer



Rechner

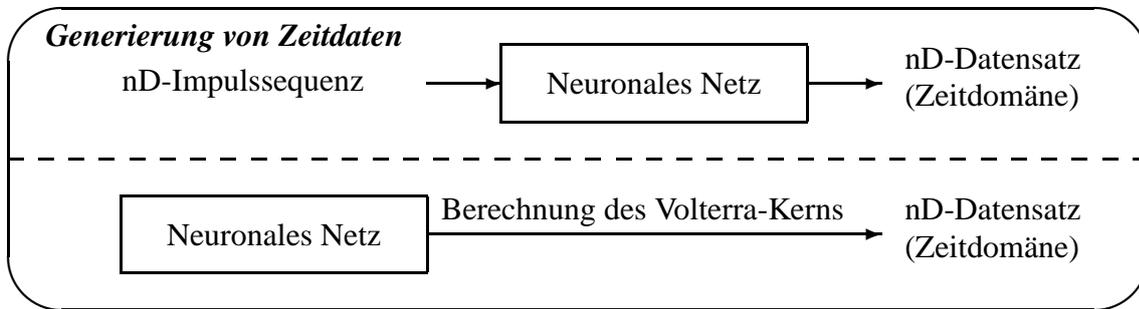
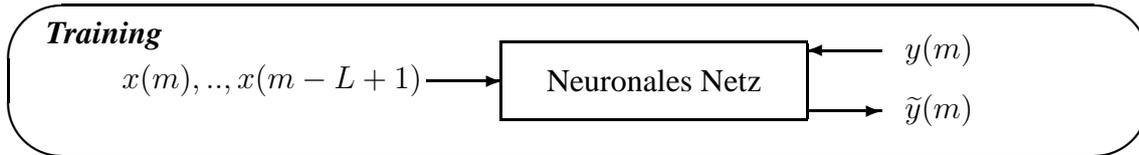


Abb. 2.8: Generierung von nD -NMR-Datensätzen (Zeitdomäne) für eine gegebene NMR-Probe.

Kippwinkeln $+\alpha$ bzw. $-\alpha$. Die Werte $+1$ und -1 treten jeweils mit der gleichen Häufigkeit p_i auf, welche von der Gedächtnislänge des Spinsystems L , der Anzahl der Impulsphasen n und dem Kippwinkel α abhängig ist. Die optimale Anregungssequenz wird generiert, indem die Impulshäufigkeit über die empirische Formel

$$p_i = \frac{450}{n \cdot \alpha \cdot L} \quad \text{mit } i = 1; -1 \tag{2.38}$$

festgelegt wird. Bei Quadraturdetektion wird die Zahlenfolge $0, +1, -1, i, -i$ eingesetzt. Mit i wird die Phase 90° kodiert, mit $-i$ die Phase 180° . Die Häufigkeit p_i aller Impulse wird ebenfalls durch Gleichung (2.38) vorgegeben ($i = 0, +1, -1, i, -i$).

2.4.2 Netztraining

Mit den aus dem ersten Schritt gewonnenen Messdaten wird ein neuronales Netz trainiert. Die Eingabeschicht umfasst L reelle bzw. komplexe Neuronen, die Ausgabeschicht stets nur ein reelles bzw. komplexes Neuron. Die Größe I der Zwischenschicht richtet sich nach der Komplexität des Spinsystems. Die Trainingsmuster werden durch Verschieben eines Fensters der Länge L , dessen Startposition zufällig ausgewählt wird, über die komplette Anregungssequenz erhalten (Abbildung (2.9)). Das Training findet nach dem in Abschnitt 2.3.2 beschriebenen *backpropagation*-Algorithmus statt. Das trainierte Netz repräsentiert das approximierte Übertragungsfunktional $\tilde{\Phi}[x(t)]$.

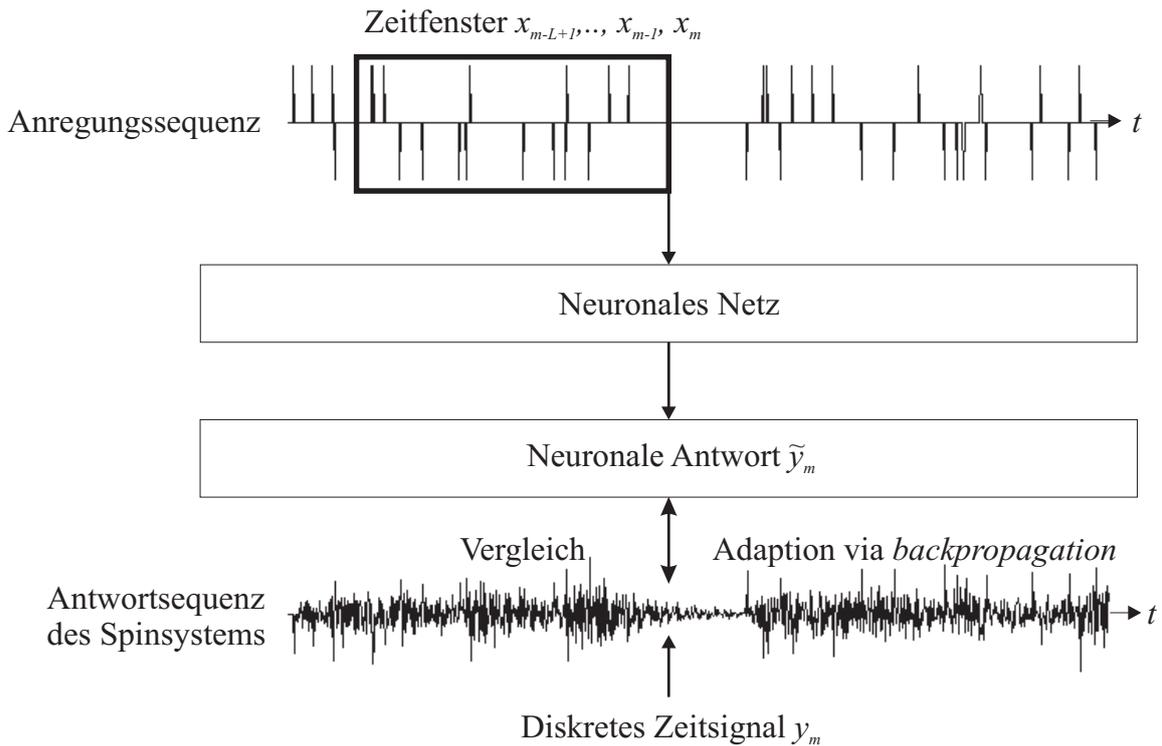


Abb. 2.9: Schema zum Training des neuronalen Netzes.

2.4.3 Generierung von Zeitdaten

Zur Generierung der Zeitdaten stehen zwei Verfahren zur Verfügung. Mit konventionellen Anregungssequenzen kann die Systemantwort berechnet werden. Zur Gewinnung eines eindimensionalen Zeitdatensatzes wird mit einer einfachen Sequenz die Antwort berechnet

$$\begin{aligned}
 \text{FID: } y(t=0) &= \tilde{\Phi}[0, 0, \dots, 0, 0, 0, 1] \\
 y(t=\Delta t) &= \tilde{\Phi}[0, 0, \dots, 0, 0, 1, 0] \\
 y(t=2 \cdot \Delta t) &= \tilde{\Phi}[0, 0, \dots, 0, 1, 0, 0] \\
 &\vdots \\
 y(t=(L-1) \cdot \Delta t) &= \tilde{\Phi}[1, 0, \dots, 0, 0, 0, 0],
 \end{aligned}$$

welche $\tilde{\Phi}[\delta(t)]$ entspricht. Bei komplexen Netzen kann zusätzlich der Imaginärteil des FIDs berechnet werden. Die Zeitdaten eines konventionellen COSY-Experimentes lassen sich wie folgt über $\tilde{\Phi}[\delta(t_1) + \delta(t_2); t_2 \leq t_1]$ berechnen

$$\begin{aligned}
 1. \text{ FID: } y(t_1 = \Delta t, t_2 = 0) &= \tilde{\Phi}[0, 0, \dots, 0, 0, 1, 1] \\
 y(t_1 = \Delta t, t_2 = \Delta t) &= \tilde{\Phi}[0, 0, \dots, 0, 1, 1, 0] \\
 &\vdots \\
 2. \text{ FID: } y(t_1 = 2\Delta t, t_2 = 0) &= \tilde{\Phi}[0, 0, \dots, 0, 1, 0, 1] \\
 y(t_1 = 2\Delta t, t_2 = \Delta t) &= \tilde{\Phi}[0, 0, \dots, 1, 0, 1, 0] \\
 &\vdots \\
 &\vdots
 \end{aligned}$$

Zur Axialsignalunterdrückung wird wie bei der konventionellen Technik eine zweite Antwort mit einer um 180° gedrehten Phase (Kodierung -1) des zweiten Impulses berechnet und zur ersten Antwort summiert. Unterdrückung von Null- und Doppelquantenkohärenzen sind ebenfalls durch Phasenzyklen möglich.

Die entstandenen Zeitdaten entsprechen denen in Abschnitt 2.1.3 konventionell gewonnenen. Daher ist die Weiterverarbeitung der berechneten Zeitdaten – ob reell oder komplex – zu Frequenzdaten analog der konventionellen Verfahrensweise.

Ein alternativer Weg zur Gewinnung eines 2D-Spektrums besteht darin, die Elemente des Volterra-Kerns zu berechnen. Der Volterra-Kern selbst ist aus den Koeffizienten der Gewichtsmatrizen über

$$k_{1l} = \sum_{i=1}^I v_i c_{il} \quad (2.39)$$

$$k_{3\ l_1, l_2, l_3} = \sum_{i=1}^I v_i c_{il_1} c_{il_2} c_{il_3} . \quad (2.40)$$

erhältlich [10, 40]. Der zweidimensionale Schnitt aus k_3 kann effizient aus einem pseudo 2D-Datensatz k'_2 , dessen Fouriertransformierte das gewünschte Spektrum ergibt, über

$$k'_{2\ l_1, \Delta l_{23}} = \begin{cases} \sum_{l_2=1}^L k_{3\ l_1, l_2, l_3} & \text{mit } \Delta l_{23} = l_2 - l_3 = 0 \\ \sum_{l_2=1}^{L-\Delta l_{23}} 2k_{3\ l_1, l_2, l_3} & \text{mit } \Delta l_{23} = 1, 2, \dots, L-1 \end{cases} \quad (2.41)$$

berechnet werden. Die auf diese Weise gewonnenen Spektren enthalten weniger NN -systematische Störspuren als die berechneten Spektren aus der Abfrage über die $COSY$ -Sequenz. Es werden im Gegensatz zum approximierten Übertragungsfunktional $\tilde{\Phi}[x(t)]$, dessen Fehler sich kohärent summieren, durch die Summation in Gleichung (2.40) die relativen Fehler im Sinne einer Mittelung reduziert.

Bei der Verwendung von neuronalen Netzen mit Separation der Ordnungen – diese liefern mit dem Abfrageverfahren qualitativ schlechte Spektren – findet die Berechnung der Kerne in Abhängigkeit ihrer Transferfunktion statt [11]. Für ein komplexes Netz gilt

$$k_{1l} = v_1 c_{1l} \quad \text{nur das Neuron mit } \mathfrak{F}(z) = z \quad (2.42)$$

$$k_{3\ l_1, l_2, l_3} = \sum_{i=2}^I v_i c_{il_1} c_{il_2}^* c_{il_3} \quad \text{restlichen Neuronen mit } \mathfrak{F}(z) = z^3 . \quad (2.43)$$

Der eindimensionale Schnitt $S(\omega) = K_3(\omega, -\omega, \omega)$ weist schmalere Linien auf als die des Einimpulsexperimentes. Dieses lässt sich durch die Analyse des Schnitts $k_{3\ l, l, l}$ des Kerns dritter Ordnung zeigen. Für ein neuronales Netz, das die erste und dritte Ordnung berücksichtigt, gilt

$$k_{3\ l, l, l} = v_1 c_{1l} c_{1l}^* c_{1l} \quad (2.44)$$

$$= v_1 e^{-(\beta+i\omega)l\Delta\tau} \cdot e^{-(\beta-i\omega)l\Delta\tau} \cdot e^{-(\beta+i\omega)l\Delta\tau} \quad (2.45)$$

$$= v_1 e^{-(3\beta+i\omega)l\Delta\tau} . \quad (2.46)$$

Der Vergleich mit dem Kern erster Ordnung

$$k_{1l} = v_1 c_{1l} = v_1 e^{-(\beta+i\omega)l\Delta\tau} \quad (2.47)$$

zeigt, dass durch den Faktor $\beta = 1/(3T_2)$ die Schwingungen beim Kern erster Ordnung schneller abklingen, als beim eindimensionalen Schnitt durch den Kern dritter Ordnung. Dieses hat beim Spektrum des Einimpulsexperimentes breitere Linien zur Folge.

2.5 Spezielle Selektionsverfahren

2.5.1 Digitale Filterung

Für komplizierte Spinsysteme bietet sich die Selektion relevanter spektraler Teilbereiche durch digitale Filterung an, um die Größe des benötigten neuronalen Netzes zu begrenzen. Dies führt zu Spektren vom *semisoft-COSY*-Typ [11]. Das neuronale Netz muss hier nur einen Teil des Spinsystems lernen, sodass die Zahl der zu bestimmenden Netzkoeffizienten und somit der Rechenaufwand klein gehalten werden kann.

Die Antwortwerte $y(t)$ aus der in Abschnitt 2.4.1 beschriebenen Messung werden fouriertransformiert. Der derart entstandene Frequenzdatensatz $Y(\omega)$ wird mit einer Rechteckfunktion $H(\omega)$ multipliziert, um den gewünschten Frequenzbereich $\Delta\Omega_{\text{relevant}}$ zu selektieren

$$H(\omega) = \begin{cases} 1 & \omega \in \Delta\Omega_{\text{relevant}} \\ 0 & \text{sonst} \end{cases} \quad (2.48)$$

$$\mathcal{Y}(\omega) = H(\omega) \cdot Y(\omega) . \quad (2.49)$$

Eine inverse Fouriertransformation ergibt aus dem gefilterten Spektrum $\mathcal{Y}(\omega)$ die neuen Antwortwerte $\mathcal{Y}(t)$. Die Modulation der Schwingungen wird durch die Frequenzselektion nicht beeinträchtigt. Die gefilterten Antwortwerte $\mathcal{Y}(t)$ und die unveränderten Anregungswerte $x(t)$ werden einem neuronalen Netz zum Training präsentiert.

Das aus einem trainierten Netz berechnete 2D-Spektrum ist dem an der Diagonalen gespiegelten *semisoft-COSY* gleich. Die Selektion erfolgt in der ω_2 -Ebene, d.h. das Spektrum zeigt nur Signale in $\omega_2 \in \Delta\Omega_{\text{relevant}}$; ω_1 bleibt unverändert. Anhand der Kreuzsignale lassen sich koppelnde Kerne identifizieren.

Für weitere relevante Bereiche kann ausgehend von $y(t)$ ein weiteres $\mathcal{Y}(t)$ berechnet werden. Eine zusätzliche Messung ist nicht erforderlich.

2.5.2 Verwendung von Vorinformationen

Ist das 1D-Spektrum des Spinsystems bereits bekannt, wird das neuronale Netz modifiziert [11]. Mit J bekannten Resonanzfrequenzen Ω_j ($j = 1, \dots, J$) wird die Anregungssequenz $x(t)$ mit einem Bandpassfilter $h_j(t)$ gefiltert. Der Filter $h_j(t)$ ist die inverse Fouriertransformierte der Absorptionslinie $H_j(\omega)$ bei Ω_j . Die J gefilterten Ausgaben werden dem neuronalen Netz als Eingabevektoren präsentiert. Das neuronale Netz besitzt J Eingabe- und verdeckte Neuronen (Abbildung. (2.10)) und benutzt die Transferfunktion $\mathfrak{F}(z) = ze^{-z^2}$.

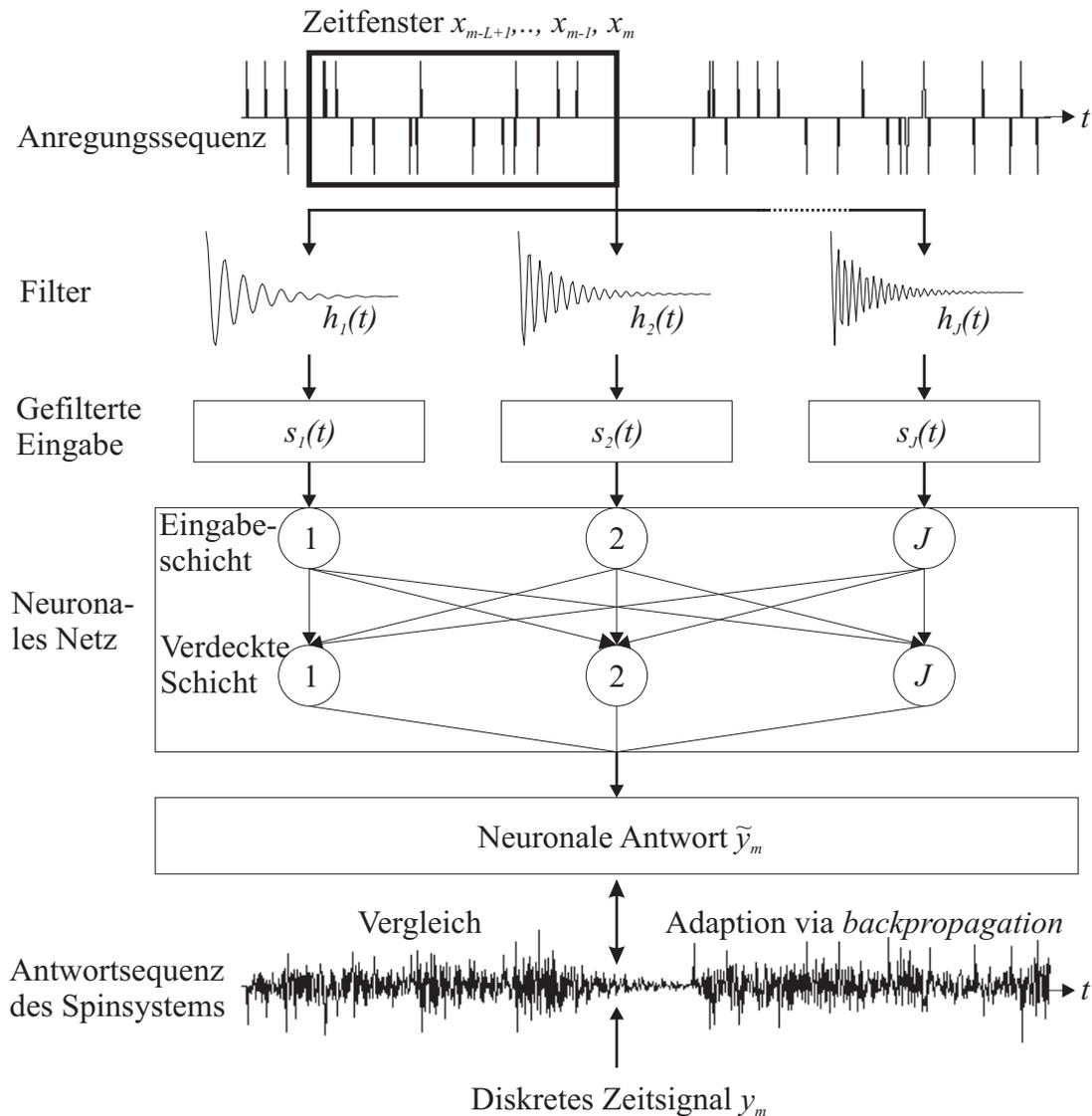


Abb. 2.10: Modifiziertes neuronales Netz zur Berücksichtigung von Vorinformationen.

In diesem Fall resultiert eine Gewichtsmatrix C in der Zwischenschicht der Größe $J \times J$, welche im Gegensatz zu der Gewichtsmatrix aus den vorhergehenden Abschnitten der Größe $L \times I$ signifikant kleiner ist². Ausschlaggebend für die Größe von C ist nun die Anzahl der Resonanzfrequenzen. Die Anzahl der Gewichte kann somit auf die q -te Wurzel minimiert werden, wobei

$$q = \frac{\ln(L \times I)}{\ln(J \times J)} \quad (2.50)$$

gilt. Für ein AX+P-Spinsystem ist $q \approx 3$. So hat das neuronale Netz weniger Gewichte anzupassen und die Rechenzeit verkürzt sich entsprechend.

Das mit Vorinformationen trainierte neuronale Netz kann das 2D-Spektrum nur mit Hilfe des Volterra-Kerns k_3 berechnen. Die Abfrage des Netzes mit der COSY-Sequenz liefert keine

²Die Größe L des Gewichtsvektors \vec{v} wird vernachlässigt.

brauchbaren Resultate.

Eine genaue Analyse der Gewichtsmatrix C zeigt interessante Eigenschaften. Die Gewichte c_{il} haben einen verschwindenden Betrag, wenn die Resonanzlinien bei Ω_i und Ω_l nicht zum selben koppelnden Spinsystem gehören oder eine verschwindende Fernkoppelung besitzen. C ist symmetrisch, da aufgrund der kommutativen Eigenschaften von koppelnden Spinsystemen $c_{il} = c_{li}$ gilt. Betragsmäßig große Nichtdiagonalelemente spiegeln koppelnde Resonanzlinien wider. Ein Berechnen der mehrdimensionalen Zeit- und Frequenzdatensätze sowie ein Darstellen der Spektren zur Analyse ist für diese Art von Netzen daher nicht zwingend notwendig.

2.6 Bewertungskriterium für alternative NN-Verfahren

Die in dieser Arbeit verwendeten Netze werden mit neuen Algorithmen oder neuen Ansätzen auf der Grundlage bekannter Daten trainiert. Um eine quantitative Aussage treffen zu können, wird der lineare Korrelationskoeffizient r zweier Stichproben nach

$$r = \frac{\sum_{n=1}^N a_n b_n}{\sqrt{\sum_{n=1}^N a_n^2} \sqrt{\sum_{n=1}^N b_n^2}} \quad (2.51)$$

berechnet, wobei die N Daten im Vektor \vec{a} als Referenz betrachtet werden und \vec{b} die zu überprüfenden Daten beinhaltet. Sind die beiden Datensätze äquivalent, so nimmt r den Wert 1 an. Die Größe von N sollte nicht zu klein sein, da sonst aus statistischen Gründen ungenaue Korrelationskoeffizienten resultieren.

Bei der mehrdimensionalen NMR-Spektroskopie ist es von Vorteil, den 1D-Anteil aus dem Datensatz zu entfernen. Durch Subtraktion nach

$$y_{m,\text{sub}} = y_{m,\text{gesamt}} - \sum_{l=1}^L f_l d_l \cdot x_{m-l} \quad (2.52)$$

mit $m = 1, \dots, M$ und $x_{m-l} = 0$ für $m - l < 0$

können so die Beiträge der Impulse entfernt werden. Wird der Korrelationskoeffizient nach Gleichung (2.51) berechnet, so muss beiden Datensätzen \vec{a} und \vec{b} der lineare Anteil abgezogen werden.

Zum Vergleich der Daten wäre auch die Bildung des einfachen quadratischen Mittels denkbar. Dieser zeigt jedoch gegenüber dem Korrelationsverfahren keine Möglichkeit lineare Beziehungen der Art $r_s = a + bx_s$ direkt bestimmen zu können. Die Bildung des quadratische Mittels nach Entfernen des 1D-Anteils aus dem Datensatz ist wenig sinnvoll, da aufgrund einer fehlenden Referenz die Daten nicht skaliert werden können.

3 Programme und Apparaturen

Die in dieser Arbeit verwendeten Programme sind weitestgehend selbst entwickelt oder modifiziert, um für die NMR-Signalanalyse akzeptable Resultate zu erlangen. Dieses erfordert auch einen Eingriff in bereits lauffähige Software (z.B. *GAMMA* oder *SNNS*). Das Studium und die Anpassung dieser Programme wird durch den öffentlichen Quellcode ermöglicht. Es sind im folgenden die wichtigsten Programme vorgestellt. Anschließend werden die Modifikationen beschrieben, die sich aus der experimentellen Fragestellung ergeben.

3.1 Verwendete Programme für die NMR-Simulationen

Anstelle von experimentellen Daten, die von unerwünschten Artefakten wie schwankenden Hf-Leistungen, Phasenfehler oder Inhomogenitäten des Magneten begleitet werden, kommen vor allem Simulationsrechnungen zum Einsatz.

3.1.1 *SPECSI*

Das Simulationsprogramm *SPECSI* berechnet auf der Grundlage des Dichtematrixformalismus die Antwort eines NMR-Spinsystems auf eine gegebene Anregung [41]. Es ist in der Programmiersprache *C* geschrieben und stellt dem Anwender eine Sammlung von Kommandos zur Generierung und Manipulation von Spinsystemen zur Verfügung. Das Programm erwartet die Angaben von Frequenzen, der Koppelungskonstanten, der Relaxationszeit und des Abtastintervalls. Die Datengenerierung erfolgt über Kommandos zur Steuerung von Impulsen und freien Evolutionszeiten. Durch den Einsatz von Zusatzprogrammen, welche die benötigten Einzelkommandos für *SPECSI* erzeugen, kann die Simulation von beliebig komplexen Experimenten erfolgen.

Ausgehend von der Dichtematrix im thermischen Gleichgewicht σ_{eq} können Hf-Impulse mit der Phase φ und dem Kippwinkel α durch eine Drehung der Dichtematrix $\sigma(t)$ in $\sigma_\alpha(t)$ approximiert werden¹

$$\sigma_\alpha(t) = e^{-\mathbf{I}\varphi\alpha} \sigma(t) e^{\mathbf{I}\varphi\alpha}. \quad (3.1)$$

Die Spin-Gitter-Relaxation T_1 und die Spin-Spin-Relaxation T_2 wird durch einen phänomenologischen Ansatz während der freien Evolution für die einzelnen Elemente der Dichtematrix berücksichtigt. Wird die Dichtematrix $\sigma(t)$ in der Eigenbasis des Hamiltonoperators geschrieben, können Relaxationsprozesse durch eine exponentielle Annäherung an den Gleichgewichtszustand beschrieben werden. Es gilt für die Diagonalelemente σ^{ii}

$$\sigma^{ii}(t + \Delta t) = \sigma_{eq}^{ii} - (\sigma_{eq}^{ii} - \sigma^{ii}(t)) e^{-\frac{\Delta t}{T_1}} \quad (3.2)$$

¹Für unendlich kurze Impulse gilt Gleichung 3.1 exakt.

und für die Nichtdiagonalelemente σ^{ij}

$$\sigma^{ij}(t + \Delta t) = \sigma^{ij}(t) e^{-\frac{\Delta t}{T_2}}. \quad (3.3)$$

Bei diesem einfachen Ansatz wird angenommen, dass die Relaxationszeiten aller Einzelspins gleich sind.

Mit Hilfe von *SPECSI* sind die Spinsystemantworten von allen im Anhang A.1 aufgeführten Systemen berechnet worden.

3.1.2 GAMMA

Eine auf *C++* basierende Programmbibliothekssammlung zur quantenmechanischen Simulation von NMR-Phänomenen ist *GAMMA* [42]. Es liegt derzeit in der Version 4.0.3 vor [53, 54].

Mit Hilfe von geeigneten Funktionen aus der *GAMMA*-Bibliothek können ausführbare Programme geschrieben werden. Damit die Simulationen von mehrdimensionalen Experimenten in einer angemessenen Zeit durchführbar sind, verzichtet *GAMMA* i. Allg. auf Relaxationseffekte. Die zu detektierenden Signale werden mit einem Exponentialfaktor multipliziert, um ein abklingendes Signal zu erhalten. Funktionen zur Behandlung von stochastischen Impulssequenzen stellt *GAMMA* nicht zur Verfügung. Die von einer beliebigen Anregungssequenz abhängige Detektion in diskreten Zeitabständen kann mit Hilfe von *GAMMA* nur unter Verwendung von Superoperatoren realisiert werden. Da die für die NMR-Spektroskopie mittels neuronaler Netze verwendeten Anregungssequenzen in der Größenordnung von 10^5 bis 10^6 liegen, wird selbst für kleine Spinsysteme eine enorme Rechenkapazität benötigt. Somit wurde *GAMMA* nur für Kontrollsimulationen von 1D- und 2D-Experimenten verwendet.

3.1.3 ACD/HNMR

Das kommerzielle Programm *ACD/HNMR* der Firma *Advanced Chemistry Development Inc.* [55] kann chemische Verschiebungen sowie Koppelungsparameter für Strukturformeln auf der Basis einer programminternen Spektrendatenbank abfragen. Kommt ein Strukturelement vor, das nicht von der Datenbank erfasst ist, so versucht das Programm ähnliche Strukturen in der Datenbank zu ermitteln und an das gegebene Strukturelement anzupassen. Ist dieses nicht möglich, so werden schließlich empirische Regeln angewandt, um die nötigen Parameter abzuschätzen.

In dieser Arbeit wurde die Version 1.0 verwendet. Die berechneten NMR-Parameter finden sich im Anhang A.1.5 wieder.

3.2 Verwendete Programme für neuronale Netze

3.2.1 Spezielle Programmcodes zur NMR-Signalanalyse

Ein in der Programmiersprache *C* geschriebener Code wurde speziell für die in der NMR-Signalanalyse benötigten neuronalen Netze erstellt. Das Erzeugen und Initialisieren der Gewichtsmatrizen für das neuronale Netz, das Training, sowie die Abfrage des trainierten Netzes

nach konventioneller Art oder mit Hilfe des Volterra-Kerns fanden dabei in separaten Programmen statt. Da die Simulationen, die diese Programme ausführen, extrem rechenintensiv sind, kam es bis heute zu keinem Einsatz im Routinemessbetrieb. Es wurden daher mögliche Zeitoptimierungen untersucht und vorgenommen. Mit Hilfe einer Verteilung auf mehrere Rechner-systeme können variable *NN*-Parametersätze auf ihre Tauglichkeit getestet werden, da ein guter Satz selten a priori bekannt ist. Somit konnten zusätzliche Zeitersparnisse erreicht werden. Die technischen Details finden sich im Anhang A.2.

3.2.2 SNNS

Der *Stuttgarter Neuronale Netzwerk Simulator (SNNS)* [43] ist ein in *C* geschriebener Softwaresimulator für neuronale Netze und wurde am *Institut für Parallele und Verteilte Höchstleistungsrechner (IPVR)* an der Universität Stuttgart entwickelt. Er ist unter [56, 57] erhältlich und für den wissenschaftlichen Einsatz frei verfügbar.

Mit Hilfe des *SNNS* (Version 4.2 [44]) sind simulierte NMR-Systeme untersucht worden, da durch die Vielzahl der verschiedenen Lernalgorithmen und Netztopologien für neuronale Netze sich weitere Möglichkeiten für den Einsatz in der NMR-Signalanalyse ergeben könnten. Um Vergleichsmöglichkeiten zu den eigenen neuronalen Netzprogrammen zu haben, ist der *SNNS* um die entsprechenden Transferfunktionen $\mathfrak{f}(z) = ze^{-z^2}$, und $\mathfrak{f}(z) = z^3$ ergänzt worden.

Da dieser Simulator nicht speziell für den Einsatz in der NMR-Signalanalyse programmiert, sondern vielmehr für eine Vielzahl von Problemstellungen konzipiert wurde, sind Optimierungen in der Programmstruktur aufgrund eines erhöhten Aufwands wenig sinnvoll. Einzig die in Anhang A.2.1 beschriebenen Compilerschalter wurden getestet um evtl. Optimierungen vornehmen zu können. Es sind jedoch mit diesen Compilerschaltern Instabilitäten des Programms aufgetreten, sodass teilweise Algorithmen nur im nicht optimierten Code oder im Defektmodus (*debug*) lauffähig waren, was erhebliche Rechenzeiten zur Folge hatte. Hinzu kommen aufgrund der *ASCII*-Formatverarbeitung von *SNNS* erhebliche Anforderungen an Arbeitsspeichergröße des Rechners. Daher musste die Analyse auf einfache Spinsysteme beschränkt bleiben.

3.3 Experimenteller Teil

3.3.1 Benutzte NMR-Apparatur

Experimentelle *NN*-Daten wurden mit einem umgerüsteten Spektrometer *WM 360* der Firma *BRUKER* erhalten². Da die Leistung des Hochfrequenz-Verstärkers über die gesamte Messzeit konstant bleiben muss, wird anstelle des *F1*-Kanals der dauerlastfähige Entkoppelerkanal *F2* benutzt. Ein Rechner der Firma *Atari* ist über eine speziell konstruierte Schnittstelle mit dem Entkoppeler und dem Spektrometerrechner *ASPECT 3000* verbunden. Der PC sorgt dafür, dass die einzelnen Impulse mit definierter Amplitude und Phase zur Probe gelangen, der *ASPECT 3000* übernimmt die Steuerung der Impulslängen, der Abtastzeiten und die Aufnahme der Messdaten.

Die maximale Anzahl der Messpunkte ist durch den Arbeitsspeicher des *ASPECT 3000* limitiert. Da er einen Speicher von einem Megawort besitzt, ergibt sich somit eine maximale Aufnahmelänge von $512k$ komplexen Messpunkten und eine maximale Messdauer von $1048576\Delta t$.

²Die erforderlichen Umrüstungsmaßnahmen wurden durch Herrn Dipl. Chem. F. Baumgart vorgenommen.

Nach Abspeichern der Messwerte auf die Spektrometerfestplatte werden diese über das *open aspect data transfer interface* der Firma *Resonance Instruments* auf Rechner zur Weiterverarbeitung transferiert. Dort muss eine Konvertierung der spektrometereigenen 24-Bit-Darstellung in das Standard-Fließkommaformat erfolgen.

3.3.2 Messung

Als AMX-Spinsystem wurde handelsübliches 1-Chlor-2,4-dinitrobenzol verwendet. Es wurde eine 0,05 molare Lösung in D_6 -Aceton hergestellt, die mit Sauerstoff gespült wurde, um die Relaxationszeiten kurz zu halten.

Die Bezeichnung der Kerne können der Abbildung (3.1) entnommen werden. In der Literatur werden die chemischen Verschiebungen mit $\nu_A = 7,82$ ppm, $\nu_M = 8,38$ ppm und $\nu_X = 8,66$ ppm angegeben [45]. Die Koppelungskonstanten betragen $J_{AM} = 8,82$ Hz, $J_{AX} = 0,37$ Hz und $J_{MX} = 2,57$ Hz. Die Relaxationszeiten werden mit Hilfe der *Inversion Recovery*-Methode [46, 47] zu 2,1 s für den Kern A und zu 2,4 s für die Kerne M und X bestimmt.

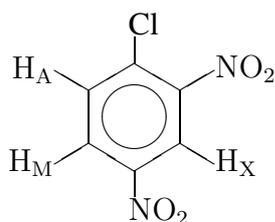


Abb. 3.1: Strukturformel des 1-Chlor-2,4-dinitrobenzol.

Zur Generierung der hier verwendeten Trainingsdatensätze gelten folgende Parameter:

Kippwinkel	15°
Phasen	0° und 180°
Impulshäufigkeit	5%
Abtastzeit	1278 μ s
Spektrale Breite	391,2 Hz
Datenpunkte	524.288
Messzeit	ca. 11 min

Vom aufgenommenen Datensatz wird der *DC*-Pegel abgezogen. Nach dem Transfer auf einen leistungsstarken Rechner werden die Werte so skaliert, dass sie im Intervall [-1,1] liegen. Die ersten 16.000 Datenpunkte werden für das Training des neuronalen Netzes nicht berücksichtigt.

4 Arbeiten und Resultate

Die in dieser Arbeit verwendeten neuronalen Netze sind teilweise mehrfach mit unterschiedlichen *NN*-Parametersätzen trainiert worden. Als Qualitätsmerkmal für ein erfolgreiches Training dienten die Diagonalsignale und die Multipletstrukturen in den berechneten 2D-Spektren. Die *NN*-Parameterwahl ist bisher nicht automatisierbar und auch von untergeordneter Bedeutung, da grundlegende Arbeiten im Vordergrund stehen. Als Bewertungskriterium wird der unter Abschnitt 2.6 beschriebene Korrelationskoeffizient angewendet.

4.1 Entwicklungen zur *NN*-Methodik

Ein chemometrischer Grundsatz ist die optimale Auswahl und Anwendung von Verfahren für ein gegebenes Problem. In den bisherigen Arbeiten zu diesem Thema sind stets *feedforward* Netze mit *backpropagation*-Algorithmus ohne weitere Modifizierungen im *online*-Training zum Einsatz gekommen [10, 11]. In den folgenden Abschnitten werden Variationen in der Netztopologie und im Lernalgorithmus vorgestellt.

4.1.1 Minimierung der Netzgröße

Eine dynamische Minimierung des Netzes ist möglich, indem nach einer bestimmten Trainingszeit die betragsmäßig kleinsten Gewichte aus dem Netz gelöscht werden (*pruning*-Verfahren). Mit diesem Löschvorgang wird die entsprechende Verbindung aus der Netztopologie herausgenommen. Das in Abbildung (4.1) dargestellte Verfahren basiert auf der Annahme, dass aufgrund der geringen Gewichtung dieser Verbindung ihr Beitrag zur Klassifikationsleistung des Netzes nur sehr beschränkt sein kann. Nach Löschung muss ein weiteres Training erfolgen, damit die verbleibenden Gewichte nachjustiert werden können. Durch das Verkleinern der anzupassenden Gewichtsmatrix resultiert je nach Anzahl der gelöschten Verbindungen eine Beschleunigung des Lernvorganges.

Bei dem angewendeten *pruning*-Verfahren werden nur Gewichte aus der Zwischenschichtsmatrix C gelöscht. Dabei werden die Gewichte auf Null gesetzt, die einen bestimmten Schwellenwert nicht überschreiten. Der Schwellenwert wird über ein Maß g aus den betragsgrößten Koeffizienten berechnet.

$$c_{il} = \begin{cases} 0 & \text{für } |c_{il}| < g \cdot \max_{i,l} |c_{il}| \text{ mit } 0 \leq g \leq 1 \\ c_{il} & \text{sonst.} \end{cases} \quad (4.1)$$

Durch diese Maßnahme wird die Gewichtsverbindung von dem Eingabewert x_l zum Zwischenschichtsneuron i gelöscht. Für $g = 0$ bleibt das Netz unverändert, für $g = 1$ werden alle Verbindungen bis auf die Referenz gelöscht. Beim weiteren Lernprozess werden die gelöschten

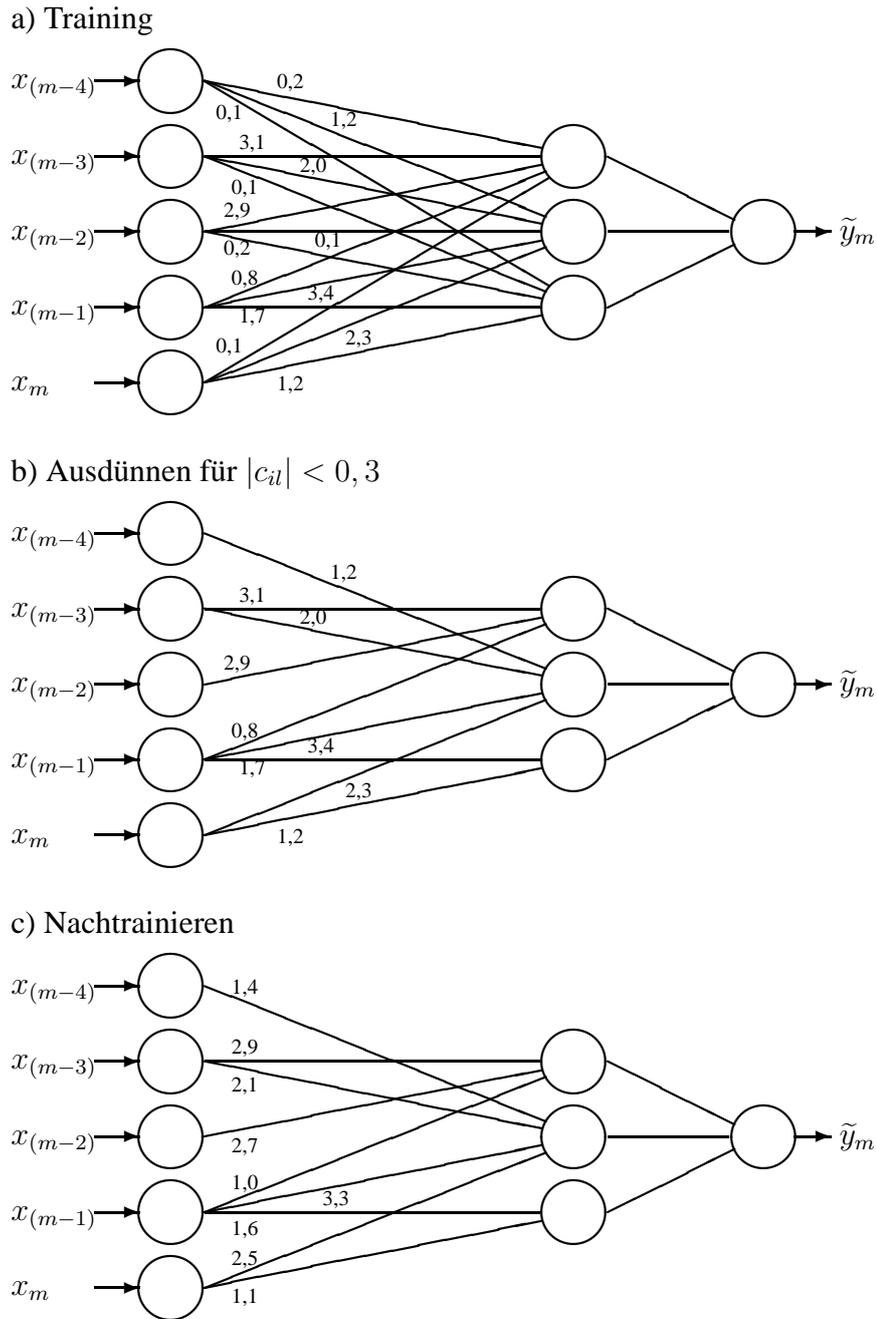


Abb. 4.1: Verkleinerung der Netztopologie nach dem *pruning*-Verfahren.

Verbindungen vom *backpropagation*-Algorithmus nicht mehr berücksichtigt.

Vor dem Löschvorgang muss gewährleistet sein, dass sich das Netz in einer konstanten Lernphase befindet. Die Herausnahme der Verbindungen kann an zwei verschiedenen Phasen im Lernprozess stattfinden: vor oder nach der Herabsetzung der Lernrate η . Beim ersten Löschschemata kann das Löschen der Verbindungen erfolgen, sobald der aktuelle mittlere quadratische Fehler größer ist, als der Mittelwert einer festgesetzten Anzahl von vorangegangenen Zyklen. Weitere Lernzyklen bei gleicher Lernrate η justieren die Gewichte nach der Modifizierung. Es folgt die Verringerung von η .

Bei dem zweiten Löschschemata wird erst die Lernrate – wie bei allen anderen Netzen – verringert und im nachfolgendem Zyklus das Schwellenwertkriterium (4.1) angewandt. Bei beiden Schemata findet vor Lernende bzw. vor dem Abbruch durch Erreichen des Konvergenzkriteriums kein Eingriff mehr in die Netztopologie statt, um den Lernerfolg nicht zu gefährden.

Für zwei verschiedene Spinsysteme ist das *pruning*-Verfahren mit unterschiedlichen Schwellenwerten untersucht worden. Alle Netze wurden mit beiden Löschschemata bis zur Erfüllung des Konvergenzkriteriums trainiert. Anschließend wurde von den trainierten Netzen über den Volterra-Kern dritter Ordnung der zweidimensionale Datensatz k'_2 berechnet. Als Referenz für die Bestimmung des Korrelationskoeffizienten $r_{k'_2}$ diente ein nicht modifiziertes Netz, welches mit denselben Datensätzen trainiert wurde.

Die Korrelation zeigte, dass das Löschschemata, bei der erst die Veränderung der Lernrate mit anschließender Löschung stattfindet, zu einer deutlich besseren Konvergenz führte. Ebenfalls für diese Variante sprach, dass weniger Lernzyklen bis zum Erreichen des Konvergenzkriteriums notwendig sind. Sie wurde daher im folgenden weiterverwendet.

In Abbildung (4.2) sind die Korrelationskoeffizienten eines AX+P-Systems (Spinparameter in Anhang A.1.1) für verschiedene Schwellenwertmaße aufgetragen. Weiterhin ist die Rechenzeitreduktion für ein Training auf demselben Rechner im Vergleich zu einem nicht modifizierten Netz dargestellt. Das Training aller Netze fand mit 400 Neuronen in der Eingabeschicht, 15 Neuronen in der Zwischenschicht, sowie den Lernparametern $\sigma = 0.02$ und $\eta = 0.9$ statt.

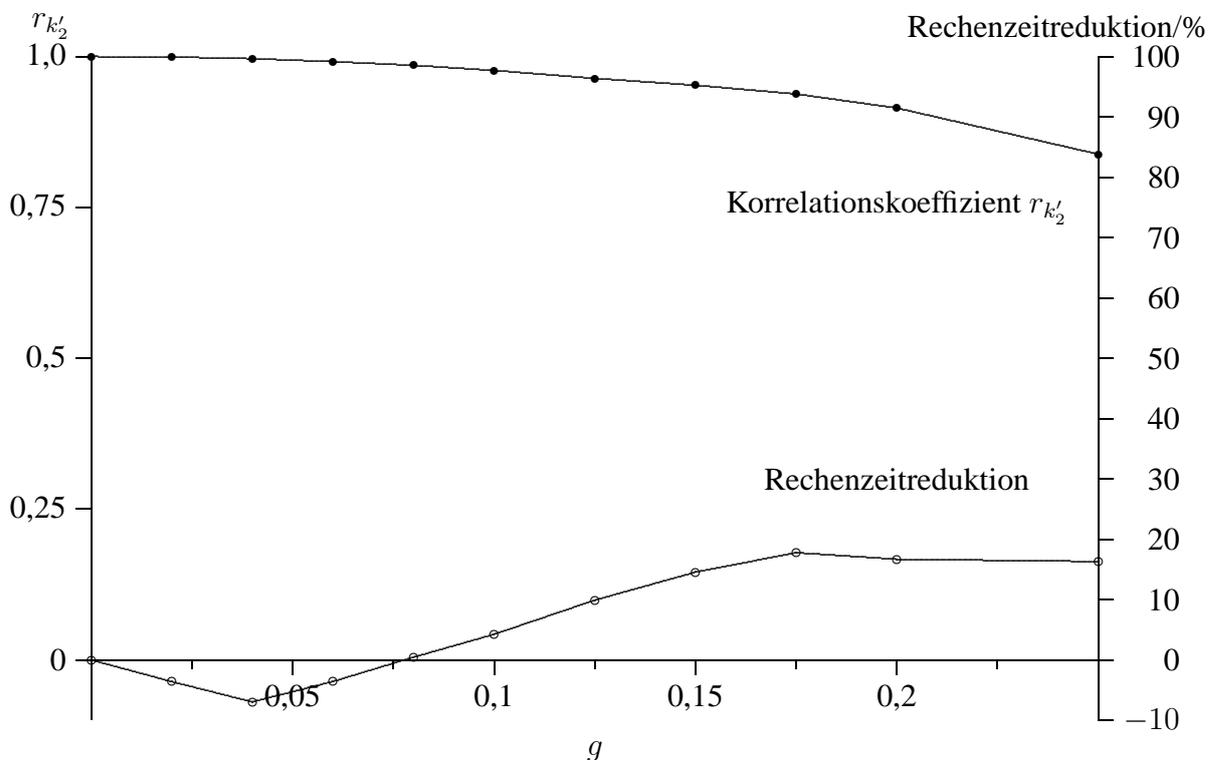


Abb. 4.2: Korrelationskoeffizient $r_{k'_2}$ und Rechenzeitreduktion für trainierte Netze eines AX+P-Spinsystems nach dem *pruning*-Verfahren gegen das Schwellenwertmaß g .

Erwartungsgemäß verschlechtert sich die Korrelation mit steigender Löschung von Gewichtsverbindungen. Proportional zu g werden Datenpunkte gemäß des Schwellenwertkriteriums auf Null gesetzt. Da diese Datenpunkte einzelne abklingende Schwingungen im k'_2 -Datensatz beschreiben, bewirken sie ein qualitativ schlechteres Spektrum. Für $g > 0,2$ lässt die Qualität des Volterra-Kerns dritter Ordnung stärker nach. Hier können die verbleibenden Gewichte die fehlenden Verbindungen nicht mehr kompensieren. Der für ein Schwellenwertmaß $g = 0,2$ berechnete 2D-Schnitt durch den Volterra-Kern dritter Ordnung ist in Abbildung (4.3) dargestellt.

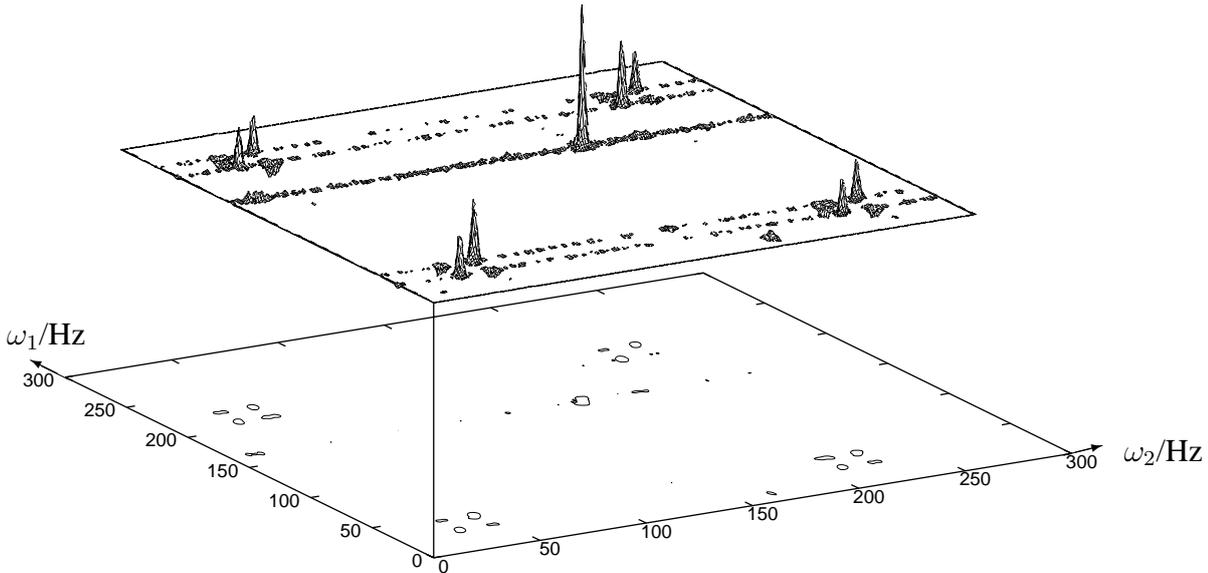


Abb. 4.3: Phasenkorrigierter 2D-Schnitt $K_3(\omega_1, \omega_2, -\omega_2)$ eines AX+P-Spinsystems für ein *pruning*-Netz, dessen Gewichtsverbindungen mit dem Schwellenwertmaß $g = 0,2$ ausgedünnt wurden.

Üblicherweise werden bei abgeschnittenen *FIDs* die fehlenden Datenpunkte geschätzt, z.B. mit Hilfe der linearen Prädiktion, welches das *FID* rekursiv extrapoliert. Da die berechneten Anfangswerte aus dem k'_2 -Datensatz bereits durch die erheblichen Verbindungs-löschungen mit Artefakten behaftet sind, werden diese durch die lineare Prädiktion reproduziert. Eine Verbesserung der Spektrenqualität wäre somit nur für geringe Werte von g möglich.

Bei kleinen Werten von g ist eine erhöhte Rechenzeit zu beobachten. Dieses liegt darin begründet, dass der Eingriff in die Netztopologie mehr Lernzyklen benötigt, bis eine Verringerung der Lernrate η stattfinden kann; das Netz konvergiert langsamer. Die gelöschten Gewichtsverbindungen können diesen Zeitverlust nicht kompensieren. Auffällig ist auch eine gleich bleibende Rechenzeitreduktion gegenüber dem nicht modifizierten Netz bei $g > 0,175$. Die Analyse des Lernverlaufs zeigt, dass das Netz viel mehr Lernzyklen bei gleich bleibender Lernrate benötigt. Dadurch bleibt die Trainingszeit konstant, obwohl deutlich weniger Gewichte anzupassen sind. Eine Abwägung von Rechenzeitreduktion zu Spektrenqualität zeigt für $g = 0,175$ den optimalen Kompromiss.

Das in Abbildung (4.4) dargestellte Diagramm zeigt den gleichen Sachverhalt für ein ABC+P-

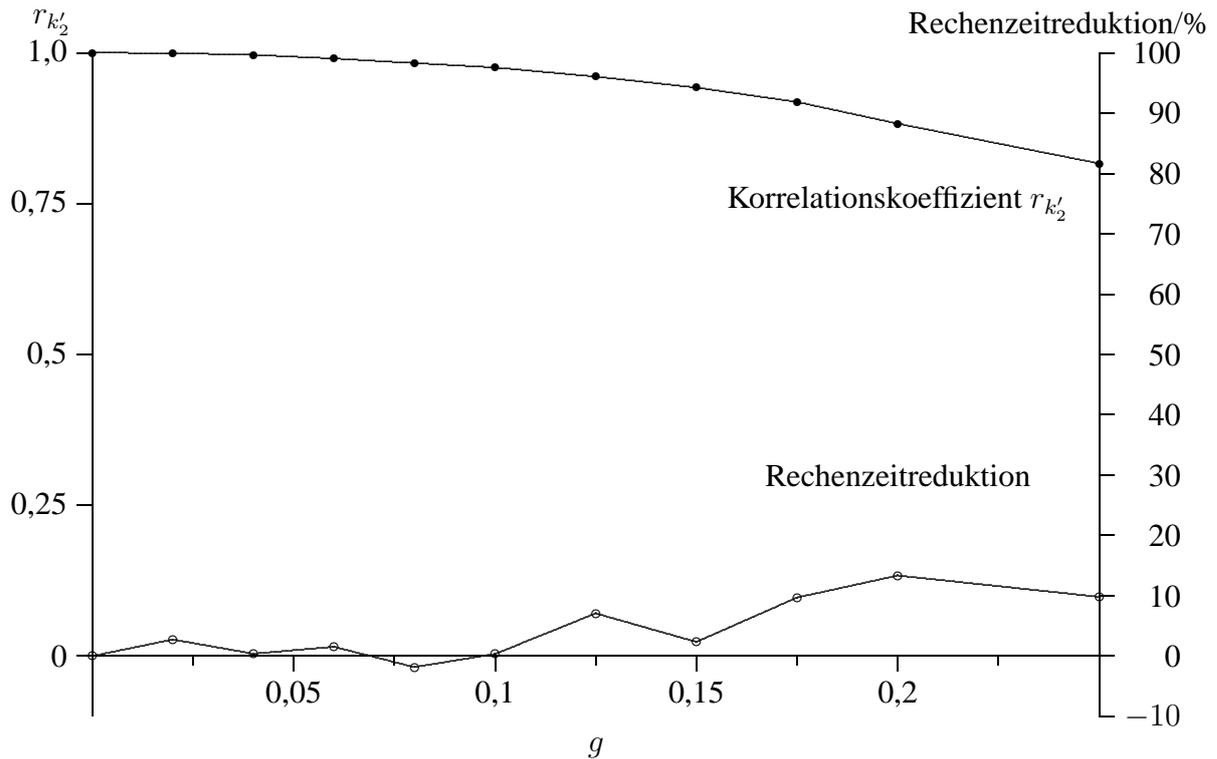


Abb. 4.4: Korrelationskoeffizient $r_{k'_2}$ und Rechenzeitreduktion für trainierte Netze eines ABC+P-Spinsystems nach dem *pruning*-Verfahren gegen das Schwellenwertmaß g .

Spinsystem. Der Verlauf des Korrelationskoeffizienten entspricht qualitativ dem des AX+P-Spinsystems, die Korrelationen sinken ein wenig stärker. Eine weniger deutliche Tendenz ist für die Kurve zu beobachten, die die Rechenzeitreduktion wiedergibt. Es findet kein Einbruch statt, jedoch kann erst für sehr große Werte von g eine klare Rechenzeitreduktion erzielt werden. Für diese Werte ist die Spektrenqualität stark gesunken und daher ein Einsatz des Verfahrens weniger sinnvoll.

4.1.2 Berücksichtigung der zeitlichen Ableitung von Anregung und Antwort

Als grundlegende Beschreibung der NN-NMR wurde

$$y_m = \sum_{i=1}^I v_i \delta [z_{i,m}] \quad \text{mit} \quad z_{i,m} = \sum_{l=1}^L c_{il} x_{m-l+1}. \quad (4.2)$$

erhalten (vergleiche Gleichung (2.33)). Das dabei zugrunde liegende Modell beschreibt das dynamische Verhalten des Spinsystems. Da die Anregungs- und die Antwortsequenz zeitlich miteinander korreliert sind, ist zu vermuten, dass das Netz diesen Zusammenhang erlernt. Die Gewichtskoeffizienten sind zeitunabhängig, daher sollten bei einem Training mit zeitlich abgeleiteten Systemdaten gleiche Koeffizienten erhalten werden. Eine Überprüfung dieses Ansatzes kann mit der Bildung der zeitlichen Ableitung getroffen werden

$$\frac{dy_m}{dt} = \frac{d}{dt} \left\{ \sum_{i=1}^I v_i \delta [z_{i,m}] \right\} \quad (4.3)$$

$$= \sum_{i=1}^I v_i \cdot \frac{d}{dt} \mathfrak{F}[z_{i,m}] \quad (4.4)$$

$$= \sum_{i=1}^I v_i \cdot \frac{d\mathfrak{F}[z_{i,m}]}{dz_{i,m}} \cdot \frac{dz_{i,m}}{dt} \quad (4.5)$$

$$= \sum_{i=1}^I v_i \cdot \frac{d\mathfrak{F}[z_{i,m}]}{dz_{i,m}} \cdot \sum_{l=1}^L c_{il} \frac{dx_{m-l+1}}{dt} \quad (4.6)$$

$$\dot{y}_m = \sum_{i=1}^I v_i \cdot \mathfrak{F}'[z_{i,m}] \cdot \sum_{l=1}^L c_{il} \dot{x}_{m-l+1} \quad (4.7)$$

Die Ableitung der Transferfunktion $\mathfrak{F}'[z_{i,m}]$ kann exakt angegeben werden; der Ausdruck erscheint bereits im *backpropagation*-Algorithmus. Die Ableitung der beiden Vektoren \vec{y} und \vec{x} kann mittels einer einfachen Differenzenbildung berechnet werden. Alternativ kann die Ableitung auch mit Hilfe der Fourierintegrale

$$f(t) = \int_{-\infty}^{\infty} F(\omega) \cdot e^{i\omega t} d\omega \quad (4.8)$$

$$f'(t) = \int_{-\infty}^{\infty} i\omega \cdot F(\omega) \cdot e^{i\omega t} d\omega \quad (4.9)$$

bzw. über die diskrete Fourierreihe

$$f(t) = \sum_{-\infty}^{\infty} c_n \cdot e^{in\omega t} \quad (4.10)$$

$$f'(t) = \sum_{-\infty}^{\infty} in\omega \cdot c_n \cdot e^{in\omega t} \quad (4.11)$$

gebildet werden, wobei die Summengrenzen der Vektorlänge angepasst werden müssen. Als Beispiel sind in Abbildung (4.5) eine ternäre Anregung mit den Werten 0,+1 und -1 sowie ihre Ableitungen gemäß beider Varianten gegeben.

Die in Abbildung (4.5c) dargestellte Ableitung ist mit systematischen Fehlern behaftet, diese sind jedoch im zeitlichen Mittel gleich Null.

Die Anwendung von Gleichung (4.7) für ein neuronales Netz ist in Abbildung (4.6) dargestellt. Sie ähnelt dem bisher verwendeten üblichen neuronalen Netz. Neben der veränderten Transferfunktion ist ein zusätzlicher Eingabevektor vorhanden, der die notwendigen zusätzlichen Daten für \vec{z} repräsentiert.

Es zeigte sich, dass die über das Fouriertheorem gebildeten Ableitungen bessere Ergebnisse lieferten. Die Verwendung einer neuen initialisierten Gewichtsmatrix ist zum Training von neuronalen Netzen dieses Typs nicht möglich. Die Gewichte erzeugen aufgrund der nicht monotonen Transferfunktion $\mathfrak{F}'[z_{i,m}]$ eine zu starke Abweichung zwischen der berechneten Antwort und der Spinsystemantwort. Dieses kann nur mit sehr kleinen Lernraten η verhindert werden, was wiederum den Lernprozess hemmt. Es können aber vortrainierte Netze, die sich schon in einem Fehlerminimum befinden, weitertrainieren werden. Die modifizierten Netze wirken dann

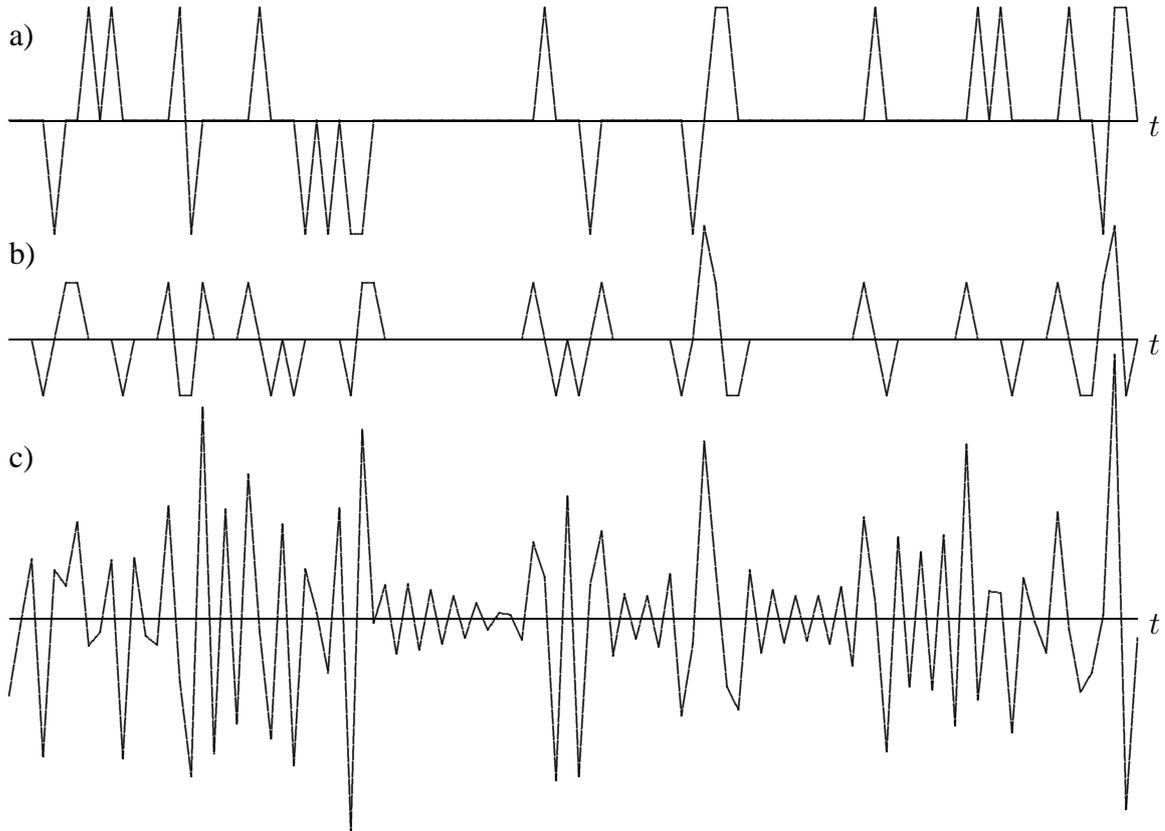


Abb. 4.5: Ausschnitt aus a) einer neuronalen Anregungssequenz und seiner Ableitung. Sie ist in b) über die Differenzenbildung und in c) mit Hilfe des Fouriertheorems berechnet worden.

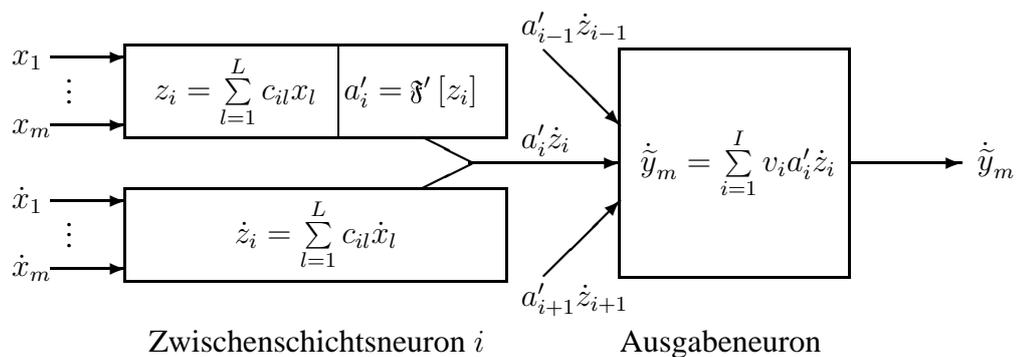


Abb. 4.6: Ausschnitt aus einem Netz für die NMR-Signalanalyse unter Berücksichtigung der Ableitungsfunktion.

als eine Art „intelligenter“ Rauschdierer, die den Gewichten einen zusätzlichen Beitrag hinzufügen, die nicht zufallsbestimmt sind. Somit kann verhindert werden, dass ein gewöhnlicher Trainingsprozess zu schnell in einem lokalen Minimum endet. Meistens ist noch ein Nachtraining mit den Standardmessgrößen bei einer leicht erhöhten Lernrate notwendig. Als negative Folge des Trainings mit den Ableitungsfunktionen ergeben sind zusätzliche Rechenzeiten von ca. 500%, da die Ableitung des Anregevektors, welcher aus Fließkommazahlen besteht, mit den

Gewichten der Zwischenschicht multipliziert werden muss.

Werden die neuronalen Netze alternierend mit den Ableitungsfunktionen trainiert, so zeigen die resultierenden Spektren kaum Unterschiede zu den rein konventionell trainierten Netzen. Bei einer stark erhöhten Anzahl von Lernzyklen können verbesserte Korrelationskoeffizienten erreicht und damit qualitativ bessere 2D-Spektren berechnet werden. Dieser Gewinn liegt jedoch im Bereich von wenigen Prozenten und steht in keinem Verhältnis zur aufgewandten Rechenzeit. Evtl. lässt sich bei Spinsystemen höherer Ordnungen eine spürbare Verbesserung erreichen. Die hier vorgestellte Entwicklung mit seiner groben Ableitungsvorschrift wäre durch *oversampling* ausbaubar, was aber weitere Rechenkapazitäten in Anspruch nehmen würde. In Abschnitt 4.2 wird ein alternativer Weg zur Behandlung komplexer Spektren aufgezeigt, der diese Vorgehensweise vollständig zu ersetzen vermag.

4.1.3 Radiale Basisfunktionsnetze

Radiale Basisfunktionsnetze wurden speziell für die Approximation mehrdimensionaler Funktionen konzipiert [48]. Die Netztopologie entspricht einer einfachen dreischichtigen *feedforward*-Architektur erster Ordnung, so wie sie bereits schon bei dem *backpropagation*-Algorithmus angewendet wurden. Vorteile der Radialnetze sind die fehlenden lokalen Minima und eine i. Allg. schnellere Trainingsphase. Nachteilig ist eine langsamere Ausführungszeit für eine Abfrage, da komplexere Berechnungen während der Klassifikationsphase durchzuführen sind.

Jedes Zwischenschichtneuron i berechnet aus einem Eingabevektor \vec{x} über die euklidische Metrik zu einem Referenzvektor \vec{r}_i und der Standardabweichung σ_i die Aktivierung $R_i(\vec{x})$

$$R_i(\vec{x}) = e^{-\left(\frac{\|\vec{x}-\vec{r}_i\|^2}{\sigma_i^2}\right)}. \quad (4.12)$$

Wird ein Muster an das neuronale Netz angelegt, werden mit Hilfe der Gaußschen Verteilungsfunktion die Neuronen aktiviert, dessen Referenzvektoren dem Eingabevektor am nächsten sind. Neuronen mit einem großen Abstand ihrer Referenz haben nahezu keine Aktivierung. Die Referenzvektoren \vec{r}_i werden auf der Basis der Trainingsmenge festgelegt. Sie besitzen die gleiche Dimension wie der Eingabevektor \vec{x} . Die Ausgabe berechnet sich aus den I Zwischenschichtneuronen mit Hilfe des Gewichtsvektors \vec{v} zu

$$\tilde{y} = \sum_{i=1}^I v_i R_i(\vec{x}). \quad (4.13)$$

Die folgenden drei Phasen bestimmen das Training.

Bestimmung der Referenzvektoren Die Vektoren \vec{r}_i – auch Stützstellen oder Zentren genannt – zeigen jene Punkte auf, durch die die zu approximierende Funktion verlaufen soll. Die Eingabe wird nach ähnlichen Mustern gesichtet und durch einen einzigen Referenzvektor repräsentiert. Durch verschiedene Clusteralgorithmen kann eine Datenreduktion erfolgen, um die Anzahl der Zwischenschichtneuronen gering zu halten. Die Bestimmung der Referenzvektoren ist der geschwindigkeitsbestimmende Schritt der Trainingsphase. Sie sollten möglichst gleichmäßig über den Musterraum verteilt sein.

Bestimmung der Standardabweichung Die Standardabweichung σ_i beschreibt die Streuung um den Referenzvektor \vec{r}_i . Bestehen Lücken im Musterraum, die durch die Referenzvektoren nicht abgedeckt sind, so können diese durch eine geschickte Wahl von σ_i geschlossen werden. Die Einstellung kann mit Hilfe der Methode des nächsten Nachbarn erfolgen. Es werden die zu \vec{r}_i nächsten Nachbarn bestimmt und daraus ein mittlerer Vektor \vec{r}'_i berechnet. Die Distanz zwischen \vec{r}_i und \vec{r}'_i ist ein Maß für die Wahl von σ_i .

Berechnung des Gewichtsvektors Dem Netz werden alle Muster präsentiert und nach dem Gradientenabstiegsverfahren iterativ der Vektor \vec{v} angepasst

$$v_i^{(t+1)} = v_i^{(t)} + \eta(y - \tilde{y}) \cdot R_i(\vec{x}). \quad (4.14)$$

Mit Hilfe von *SNNS* sind einfache Spinsysteme mit diesem Algorithmus untersucht worden. Für ein Einspinsystem zeigte sich im Vergleich zum *backpropagation*-Algorithmus eine sehr schnelle Adaption des Anregungs- und Antwortverhaltens des Spinsystems. Wie in der Literatur beschrieben, ist der aufwendigste Rechenschritt hierbei die Bestimmung der Stützstellen. Der Korrelationskoeffizient zwischen der Spinsystemantwort und der berechneten Antwort des trainierten Netzes lag bei nahezu 100% und damit höher als der für ein Netz des *backpropagation*-Algorithmus. Die Funktionalabbildung von $f[x(t)] \mapsto y(t)$ gelingt diesem Algorithmus für ein Einspinsystem außerordentlich effektiv. Ein berechnetes 2D-Spektrum lieferte das erwartete Verhalten, es waren keine Kreuzsignale sichtbar. Die gleiche Aussage kann auch für nicht koppelnde Spinsysteme getroffen werden.

Bei der Präsentation eines AX+P-Systems zum Training sank der Korrelationskoeffizient für die Reproduktion der Anregungswerte stark ab. Unbefriedigend war das Verhalten dieser Netze gegenüber der Abfrage anderer Sequenzen, wie z.B. der *COSY*-Anregung. Die Qualität der berechneten Antworten war unzureichend. Das 2D-Spektrum war stark verrauscht und wies Spuren der Signale in beiden Frequenzachsen auf. Es scheint, dass die Generalisierungsfähigkeit dieser Netze nicht den Anforderungen der NMR-Spektralanalyse entsprechen können; es fand ein reines „Auswendiglernen“ des Sachverhalts statt. Die Berechnungen der Antwort für gekoppelte Spinsysteme auf nicht trainierte Sequenzen sind für diesen Algorithmus nicht möglich.

4.1.4 Der *resilient propagation*-Algorithmus

Als Nachteil des *backpropagation*-Algorithmus haben sich die vom Benutzer festzulegenden *NN*-Parameter erwiesen. Sind sie schlecht gewählt, so kann das neuronale Netz nicht die gewünschte Approximierung erreichen. Ein Verfahren zur dynamischen Parameteranpassung ist *resilient propagation (RPROP)*, das nur auf der Basis gewichtsspezifischer Informationen Veränderungen der Neuronengewichte berechnet [49].

Die Weiterentwicklung vom *backpropagation*- zum *RPROP*-Algorithmus fand auf der Grundlage folgender Überlegungen statt. Ein Verfahren, das nur die partiellen Ableitungen an einer Stützstelle berechnet, kann nur unzureichend Informationen über die Fehlerhyperfläche widerspiegeln. Weiterhin können die optimale Schrittweite und die Schrittrichtung von Gewicht zu Gewicht unterschiedlich sein. Die Anpassung der Gewichte erfolgt daher individuell und soll

sich während des Lernprozesses dynamisch anpassen. Als Konsequenz wird nur das Ableitungsvorzeichen als Indikator für die Gewichtsanzpassung betrachtet. Die Fehlerfunktion E für den *RPROP*-Algorithmus ist gegeben durch

$$E = \sum_m (y_m - \tilde{y}_m)^2 + \mu \cdot \left(\sum_{i,l} c_{il}^2 + \sum_i v_i^2 \right) \quad (4.15)$$

wobei μ die Dämpfungskonstante für den Impulsterm ist. Das Maß der Änderung für alle Elemente c_{il} der Gewichtsmatrix wird durch einen weiteren Parameter Δ_{il} (*update-value*) bestimmt

$$c_{il}^{(t+1)} = c_{il}^{(t)} + \Delta c_{il}^{(t)} \quad (4.16)$$

$$\Delta c_{il}^{(t)} = \begin{cases} -\Delta_{il}^{(t)} & \text{für } \frac{\partial E^{(t)}}{\partial c_{il}} > 0 \\ +\Delta_{il}^{(t)} & \text{für } \frac{\partial E^{(t)}}{\partial c_{il}} < 0 \\ 0 & \text{sonst.} \end{cases} \quad (4.17)$$

Als zweiter Schritt wird $\Delta_{il}^{(t)}$ angepasst

$$\Delta_{il}^{(t)} = \begin{cases} \eta^+ \cdot \Delta_{il}^{(t-1)} & \text{für } \frac{\partial E^{(t-1)}}{\partial c_{il}} \cdot \frac{\partial E^{(t)}}{\partial c_{il}} > 0 \\ \eta^- \cdot \Delta_{il}^{(t-1)} & \text{für } \frac{\partial E^{(t-1)}}{\partial c_{il}} \cdot \frac{\partial E^{(t)}}{\partial c_{il}} < 0 \\ \Delta_{il}^{(t-1)} & \text{sonst} \end{cases} \quad (4.18)$$

mit $0 < \eta^- < 1 < \eta^+$.

Mit der Berücksichtigung der beiden partiellen Ableitungen für die zwei vorherigen Lernschritte werden Sprünge über lokale Minima durch einen Vorzeichenwechsel identifiziert; die Gewichtsänderung Δ_{il} wird um den Faktor η^- reduziert. Gleichzeitig wird im anschließenden Berechnungsschritt die Adaption ausgesetzt, um eine zu schnelle Abnahme der Schrittweite zu vermeiden. Bleibt das Vorzeichen erhalten, so wird Δ_{il} durch η^+ leicht erhöht, um die Konvergenz zu beschleunigen. Um die freie Parameterwahl einzuschränken, werden i. Allg. feste Werte für $\eta^- = 0,5$ und $\eta^+ = 1,2$ gewählt.

Im Gegensatz zum *online*-Training müssen bei *RPROP* erst alle Muster präsentiert werden, bevor eine Anpassung der Gewichte und der *update-values* erfolgen kann (*batch*-Training). Für diesen Algorithmus ist der einzig einzustellende Parameter eine Konstante μ für den Impulsterm.

Verschiedene kleine Spinsysteme wurden auf NMR-Gesichtspunkte diesem Algorithmus im *SNNS* erfolgreich präsentiert. Daher rentierte es sich, diesen Lernalgorithmus in die bereits vorhandenen eigenen Programmbibliotheken hinzuzufügen. Details dazu finden sich im Anhang A.3. Dieser Lernalgorithmus kann sowohl für die phasensensitiven und nichtsensitiven, als auch für die Netze mit Vorinformation eingesetzt werden. Für eine Vorzeichenunterscheidung wurden die entsprechenden Gleichungen (4.15) bis (4.18) auf komplexe Größen erweitert. Als Transferfunktion sollte $\mathfrak{f}(z) = ze^{-z^2}$ eingesetzt werden. Die Elemente der Gewichtsmatrix C nehmen für ein erfolgreiches Training ähnliche Werte an, wie die des *backpropagation*-Algorithmus. Eine Auswertung ist mit Hilfe der Volterra-Kerne möglich.

Eine Separation der Beiträge erster und dritter Ordnung mit Hilfe der Transferfunktionen $\mathfrak{f}(z) = z^3$ bzw. z ist nicht möglich, weil sich die Abweichungen zwischen der Spinsystem- und der approximierten Antwort mit diesen Transferfunktionen während des *batch*-Trainings zu überdimensional großen Werten summieren; das Training konvergiert nicht.

Das in Abbildung (4.7) gezeigte phasensensitive 2D-Spektrum eines AX+P-Spinsystems (Spinparameter in Anhang A.1.2) wurde auf Grundlage dieses Lernverfahrens mit der Transferfunktion $\mathfrak{f}(z) = \Re[z]e^{-\Re[z]^2} + i\Im[z]e^{-\Im[z]^2}$ für 25 Neuronen über den Volterra-Kern dritter Ordnung berechnet (Kippwinkel $\alpha = 15^\circ$, Impulstermdämpfung $\mu = 10^{-4}$, 25.000 Muster).

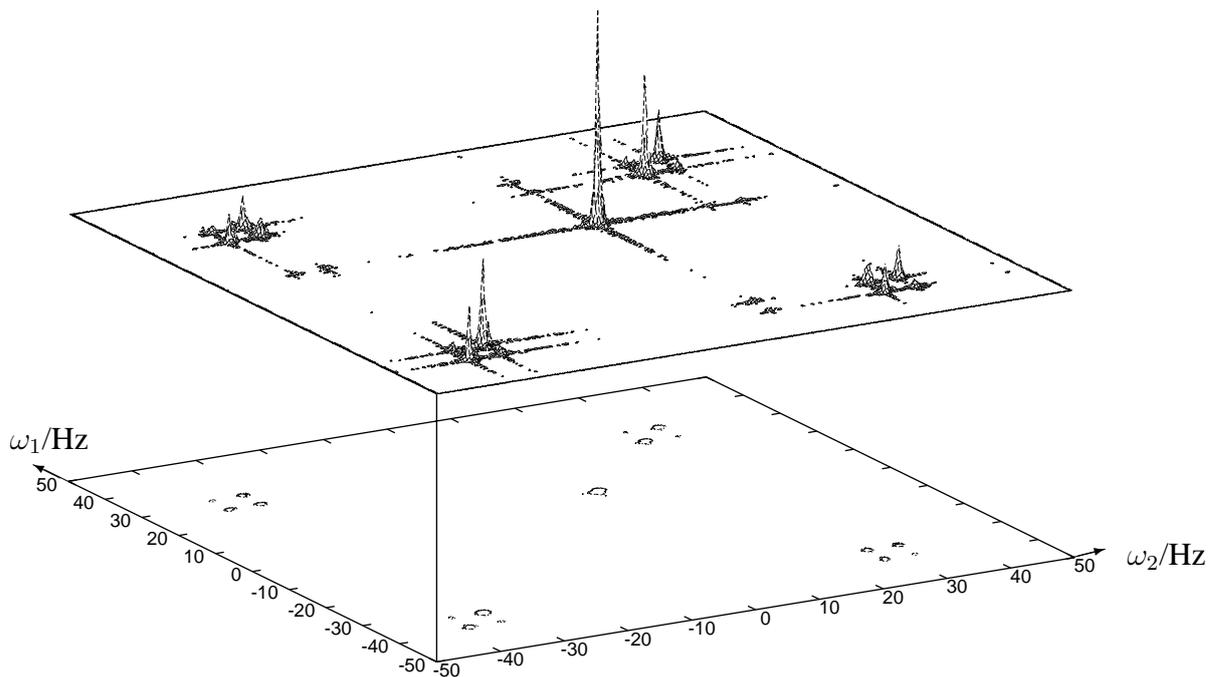


Abb. 4.7: Phasensensitiver 2D-Schnitt $K_3(\omega_1, \omega_2, -\omega_2)$ (Magnitudendarstellung) eines AX+P-Spinsystems, das mit dem *RPROP*-Algorithmus trainiert wurde.

Ein direkter Vergleich des *backpropagation*- mit dem *RPROP*-Algorithmus zeigte aufgrund des *batch*-Trainings und der einfacheren Gewichtsadaption ein ca. doppelt so schnelles Training des *RPROP*-Algorithmus pro Lernzyklus. Um auf ähnliche Korrelationen sowohl für eine reine Reproduktion der Daten, als auch für die Generierung eines 2D-Zeitdatensatzes zu kommen, hat sich die Anzahl der Lernzyklen mehr als verdoppelt. Der *RPROP*-Algorithmus erwies sich somit als rechenintensiver und langsamer. Dem ist gegenüberzuhalten, dass dieser Algorithmus sehr robust ist. Das Einstellen von nur einem nicht so kritischen – es findet keine Divergenz beim Lernen durch eine falsche Wahl statt – Wichtungsparmeter μ , der sich in kleinen Beträgen (ca. $10^{-3} - 10^{-5}$) befinden sollte, ist die große Stärke dieser Lernprozedur.

4.1.5 Der *backpercolation*-Algorithmus

Bei dem *backpercolation*-Algorithmus wird im Gegensatz zum *backpropagation* nicht der Fehler des Ausgabeneurons betrachtet, sondern der Fehler von Neuronen, die in den jeweiligen

Schichten eingebaut sind. Dadurch soll die erforderliche Anzahl der Trainingszyklen klein gehalten werden.

Für jedes Neuron in der Zwischenschicht wird der mit einem positiven Faktor λ gewichtete interne Aktivierungsfehler berechnet

$$\epsilon_i = \lambda \left(\sum_{l=1}^L c_{il} x_l - a_i \right), \quad (4.19)$$

also die Differenz aus der gewichteten Eingabe und der Ausgabe des Neurons. Der Aktivierungsfehler ϵ_i bestimmt die Änderung des Gewichtsvektors des Neurons i . Nach Erreichen eines vorgegebenen Qualitätskriteriums wird λ angepasst. Die Lernrate η kommt im *backpercolation*-Algorithmus nicht vor. Als Transferfunktion für diesen Algorithmus wird

$$\mathfrak{f}(z) = \tanh\left(\frac{z}{2}\right) \quad (4.20)$$

empfohlen [43].

Auch diesem vom *SNNS* zur Verfügung gestellten Algorithmus sind verschiedene Spinsysteme präsentiert worden. Mit einem einfachen nicht koppelnden Mehrspinsystem konnte dieser Algorithmus keine befriedigenden Resultate liefern. Die geringfügige Abnahme des mittleren Gesamtfehlers verhinderte auch die Anpassung des Magnitudenfaktors λ . Eine *COSY*-Abfrage war somit nicht möglich bzw. lieferte kein brauchbares Spektrum. Der Einsatz von anderen Transferfunktionen als (4.20) brachte keine Verbesserung im Lernprozess. Auf weiterführende Analysen von koppelnden Mehrspinsystemen wurde daher verzichtet.

4.1.6 Der *cascade-correlation*-Algorithmus

Ein grundlegendes Problem für ein erfolgreiches Training ist die Wahl der Anzahl der Neuronen in der Zwischenschicht. Mit steigender Größe des Spinsystems wird eine geeignete Wahl zunehmend kritischer, da zu wenig Neuronen die Gesamttransferfunktion nicht approximieren können, zu viele Neuronen hingegen keinen besonderen Effekt für bessere berechnete Zeitdaten erwirken und unnötige Rechenzeit erfordern. Eine mögliche Lösung bieten die *cascade-correlation*-Netze (*CC*) [50]. Hierbei wird die Topologieoptimierung besonders berücksichtigt und in den Trainingsalgorithmus integriert. Es wird mit einem minimalen Netz begonnen, das solange schrittweise vergrößert wird, bis das gewünschte Ein/Ausgabeverhalten erreicht ist. Der *CC*-Algorithmus geht wie folgt vor:

1. Die Initialisierung beginnt mit einem Netz, das nur aus den erforderlichen Eingabeneuronen und dem Ausgabeneuron besteht. Die beiden Schichten sind vollständig miteinander verbunden.
2. Beginn des Trainings mit einem gewöhnlichen Lernalgorithmus (*RPROP*, *backpropagation*, *batch-backpropagation* oder *quickprop*), bis keine Fehlerminimierung mehr stattfindet.
3. Generierung von so genannten Kandidatenneuronen, die nur mit der Eingabeschicht und der (beim ersten Durchgang noch nicht bestehenden) Zwischenschicht verbunden sind. Zwischen den einzelnen Kandidatenneuronen und zur Ausgabeschicht werden keine Verbindungen geknüpft. Die Anzahl der Kandidatenneuronen wird vorgegeben.

4. Maximierung der Korrelation zwischen der Aktivierung der Kandidatenneuronen und dem verbleibenden Fehler des restlichen Netzes. Es findet solange eine Anpassung der Gewichtsverbindungen zwischen diesen beiden Schichten statt, bis die Korrelation nicht verbessert werden kann.
5. Die Kandidatenneuronen mit den besten Korrelationen werden ausgewählt und ihre Gewichte eingefroren, d.h. diese Gewichte können anschließend nicht mehr verändert werden. Danach werden die ausgewählten Neuronen ins Netz als neue Zwischenschicht integriert. Dann beginnt der Algorithmus erneut mit Schritt 2.

Das Verfahren wird solange iterativ wiederholt, bis der Fehler unter ein vorgegebenes Konvergenzkriterium fällt.

Für ein Einspinsystem zeigte sich im *SNNS* das erwartete Verhalten. Für den *backpropagation*- und den *RPROP*-Lernalgorithmus fügte die *CC*-Prozedur ein Zwischenschichtsneuron ein, die für die NMR typischen Abfragen zeigten das gewünschte Ergebnis. Zwischen den Lernalgorithmen konnte kein Unterschied festgestellt werden. Es ist auffallend, dass für drei nicht koppelnde Spins der *backpropagation*-Lernalgorithmus drei neue Zwischenschichten einfügte, während beim *RPROP*-Lernalgorithmus zwei neue Schichten ausreichten. Die Rechenzeiten für diese einfachen Spinsysteme lagen weit über der Rechenzeit für ein normales Training mit Hilfe der *backpropagation*-Anpassung.

Für ein AX+P-Spinsystem konnte eine Konvergenz des *CC*-Algorithmus in akzeptabler Zeit nicht erreicht werden. Die Komplexität des Funktionals für dieses Spinsystem scheint zu viele neue Schichten im Netz zu fordern, sodass aus Rechenzeitgründen der Lernprozess ohne Ergebnis abgebrochen wurde, zumal das Programm aufgrund von Laufzeitproblemen ohne Optimierung rechnen musste. Eine Erhöhung der Kandidatenneuronenanzahl von den üblicherweise gewählten fünf Neuronen auf eine größere Anzahl, sowie Variationen der möglichen Lernparameter im für diese Algorithmen üblichen Bereich blieben erfolglos.

Mögliche Modifikationen des *CC*-Verfahrens, die beispielsweise versuchen die Kandidatenneuronen an verschiedenen Stellen im Netz zu generieren (*sibling/descendant cascade-correlation* [51]) oder aber teilweise schon hinzugefügte Neuronen später wieder aus dem Netz nehmen, um eine kleine Netztopologie zu gewährleisten (*pruned-cascade-correlation* [52]) konnten aufgrund der fehlenden Konvergenz nicht getestet werden.

4.2 Vorgabe von Resonanzfrequenzen

Die mit Vorgabe von Resonanzfrequenzen lernenden neuronalen Netze zeichnen sich durch ihre Schnelligkeit und ihre Fähigkeit aus mit der Gewichtsmatrix die Konnektivitäten direkt anzuzeigen. Dieses Potenzial wurde an mehreren Beispielen untersucht, um die Grenzen dieser Netze zu erkennen. Die NMR-Daten wurden hauptsächlich durch Simulationsrechnungen gewonnen, jedoch in einigen Fällen auch experimentell ermittelt (Abschnitt 4.2.6). Als Lernalgorithmus für alle folgenden Netze diente das *backpropagation*-Verfahren.

4.2.1 Trennung von überlagerten Spinsystemen ohne Überlappung

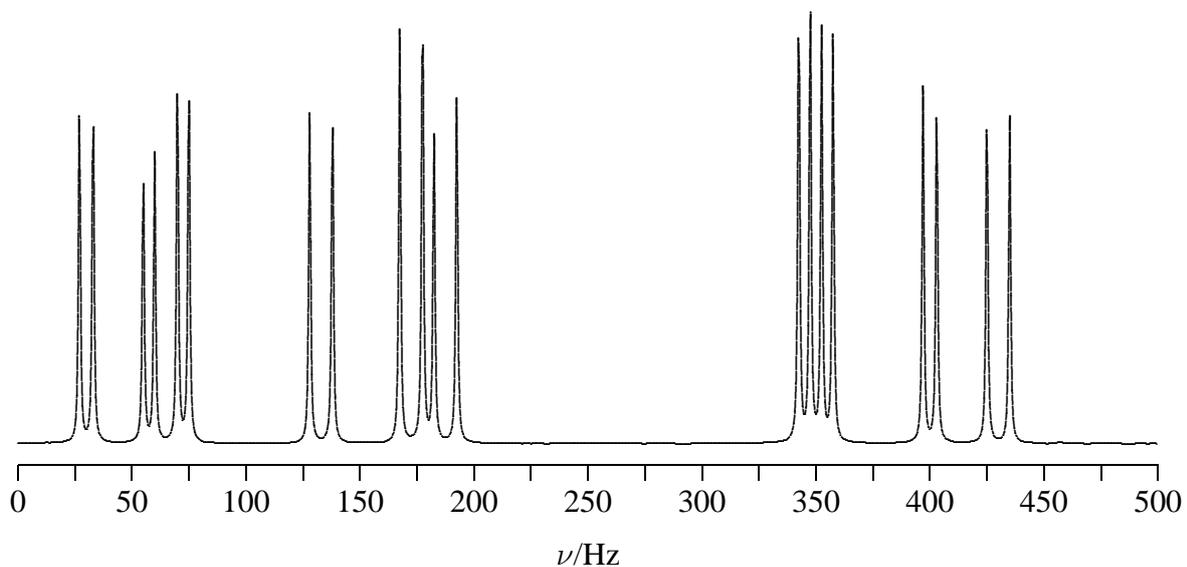


Abb. 4.8: 1D-Spektrum einer Überlagerung unbekannter Beispielsysteme.

Als Testfall hinsichtlich der Trennungseigenschaften dieser Netze werden überlagerte Spinsysteme (Spinparameter in Anhang A.1.3), dessen Resonanzen getrennt detektierbar sind, untersucht. Die 20 Frequenzen des eindimensionalen Spektrums in Abbildung (4.8) wurden verschiedenen neuronalen Netzen als Vorinformation präsentiert. Es sind Netze bei unterschiedlichen Parametern und Kippwinkeln von 15° , 30° und 45° trainiert worden. Es zeigte sich, dass die Netze bei kleinen Kippwinkeln qualitativ bessere Resultate lieferten. Generell ist festzustellen, dass bei kleinen Netzen auch kleinere Lernraten η mit kleinen Impulstermkoeffizienten μ gewählt werden sollten. Da durch die Vorgabe der gefilterten Eingabedaten die resultierende Gewichtsmatrix C weniger Spielraum bei der Anpassung hat, ist eine ungünstige Wahl des Parametersatzes weniger kritisch, als bei Netzen, die ohne Vorinformationen lernen müssen. Wichtig ist jedoch eine ausreichend lange Anregungs- und Antwortsequenz, die dem neuronalen Netz zur Anpassung präsentiert werden muss. Aus vielen Testrechnungen hat sich als geeigneter Wert eine Gesamtlänge von $100L$ erwiesen.

Die aus einem trainierten Netz resultierende Gewichtsmatrix C für den Kippwinkel $\alpha = 15^\circ$ ($\eta = 0,005$ und $\mu = 0,25$) ist in Tabelle (4.1) dargestellt. Die hervorgehobenen Werte geben die größten Beiträge einer auftretenden Resonanz wieder und zeigen deutlich die zugehörigen Konnektivitäten. Zum Vergleich ist in Abbildung (4.9a) das aus der Gewichtsmatrix via

Volterra-Kern berechnete 2D-Spektrum in Magnitudenform gegeben.

Freq.	Ω_1	Ω_2	Ω_3	Ω_4	Ω_5	Ω_6	Ω_7	Ω_8	Ω_9	Ω_{10}	Ω_{11}	Ω_{12}	Ω_{13}	Ω_{14}	Ω_{15}	Ω_{16}	Ω_{17}	Ω_{18}	Ω_{19}	Ω_{20}
Ω_1	12,2	0,0	-0,3	-0,2	-0,3	-0,5	-0,8	0,2	0,5	0,5	-0,4	-0,1	0,0	0,0	-0,1	0,3	2,1	-7,8	-0,2	1,2
Ω_2	0,0	12,0	-0,5	-0,4	-0,3	0,6	-0,9	-0,4	0,1	0,1	-0,2	-0,6	-0,3	0,4	-0,1	0,5	-3,3	3,2	-1,2	0,3
Ω_3	-0,3	-0,5	9,5	-0,5	-0,8	-1,0	0,5	-0,6	1,1	-0,3	-1,2	0,4	1,6	-2,3	0,2	-0,5	-0,5	0,2	-0,4	-0,4
Ω_4	-0,2	-0,4	-0,5	10,1	0,4	-0,1	-0,2	0,2	-0,2	1,2	-0,5	-3,5	-2,5	2,1	-0,2	-0,3	0,1	0,3	0,0	-0,5
Ω_5	-0,3	-0,3	-0,8	0,4	10,5	-0,3	0,2	-0,2	-2,6	-0,4	1,9	-0,5	0,1	-0,1	2,5	-3,8	-0,5	0,0	0,3	-0,1
Ω_6	-0,5	0,6	-1,0	-0,1	-0,3	11,7	-0,6	-0,1	-0,5	-2,7	0,0	1,7	-0,2	-0,2	-1,6	2,0	0,0	0,3	-0,6	-0,3
Ω_7	-0,8	-0,9	0,5	-0,2	0,2	-0,6	11,8	0,7	0,3	0,0	-1,1	-0,3	0,0	-0,2	0,2	0,4	0,0	0,5	2,2	-4,2
Ω_8	0,2	-0,4	-0,6	0,2	-0,2	-0,1	0,7	13,2	-0,4	0,1	0,9	0,0	-0,1	-1,4	0,0	0,2	-1,9	-0,5	-2,3	3,7
Ω_9	0,5	0,1	1,1	-0,2	-2,6	-0,5	0,3	-0,4	12,8	-0,3	-0,8	-0,2	2,5	0,1	-3,1	0,1	0,0	-0,9	0,2	0,0
Ω_{10}	0,5	0,1	-0,3	1,2	-0,4	-2,7	0,0	0,1	-0,3	12,7	0,4	-0,1	-3,1	0,1	3,4	-0,5	-0,7	-0,4	0,0	0,1
Ω_{11}	-0,4	-0,2	-1,2	-0,5	1,9	0,0	-1,1	0,9	-0,8	0,4	9,4	0,1	-0,3	2,6	-0,7	-3,5	-0,1	0,5	-0,1	-0,2
Ω_{12}	-0,1	-0,6	0,4	-3,5	-0,5	1,7	-0,3	0,0	-0,2	-0,1	0,1	12,0	0,4	-2,6	0,3	1,9	-0,3	0,4	0,0	-0,2
Ω_{13}	0,0	-0,3	1,6	-2,5	0,1	-0,2	0,0	-0,1	2,5	-3,1	-0,3	0,4	12,6	-0,1	-0,3	0,3	0,4	0,1	-0,2	-0,4
Ω_{14}	0,0	0,4	-2,3	2,1	-0,1	-0,2	-0,2	-1,4	0,1	0,1	2,6	-2,6	-0,1	12,6	-0,8	0,2	-0,3	-0,1	-0,1	0,0
Ω_{15}	-0,1	-0,1	0,2	-0,2	2,5	-1,6	0,2	0,0	-3,1	3,4	-0,7	0,3	-0,3	-0,8	11,6	0,0	-0,1	0,2	-0,7	-1,1
Ω_{16}	0,3	0,5	-0,5	-0,3	-3,8	2,0	0,4	0,2	-0,1	-0,5	-3,5	1,9	0,3	0,2	0,0	11,9	0,3	0,8	0,0	-0,7
Ω_{17}	2,1	-3,3	-0,5	0,1	-0,5	0,0	0,0	-1,9	0,0	-0,7	-0,1	-0,3	0,4	-0,3	-0,1	0,3	12,4	0,0	0,0	0,3
Ω_{18}	-7,8	3,2	0,2	0,3	0,0	0,3	0,5	-0,5	-0,9	-0,4	0,5	0,4	0,1	-0,1	0,2	0,8	0,0	12,6	0,3	-0,1
Ω_{19}	-0,2	-1,2	-0,4	0,0	0,3	-0,6	2,2	-2,3	0,2	0,0	-0,1	0,0	-0,2	-0,1	-0,7	0,0	0,0	14,1	0,3	0,3
Ω_{20}	1,2	0,3	-0,4	-0,5	-0,1	-0,3	-4,2	3,7	0,0	0,1	-0,2	-0,2	-0,4	0,0	-1,1	-0,7	0,3	-0,1	0,3	11,3

Tabelle 4.1: Gewichtsmatrix C für ein Beispielsystem aus mehreren Spinsystemen (zwei AX-Spinsysteme und ein AMX-Spinsystem).

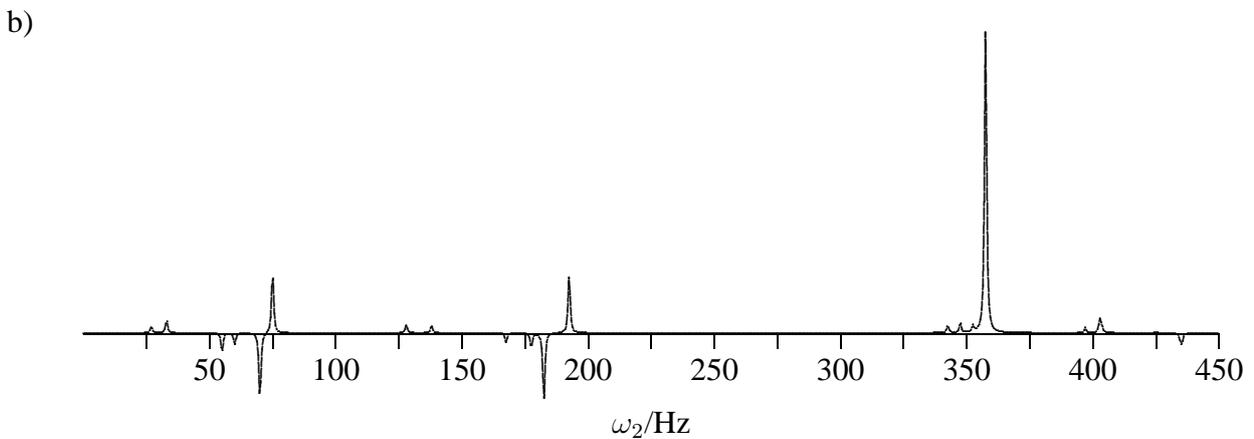
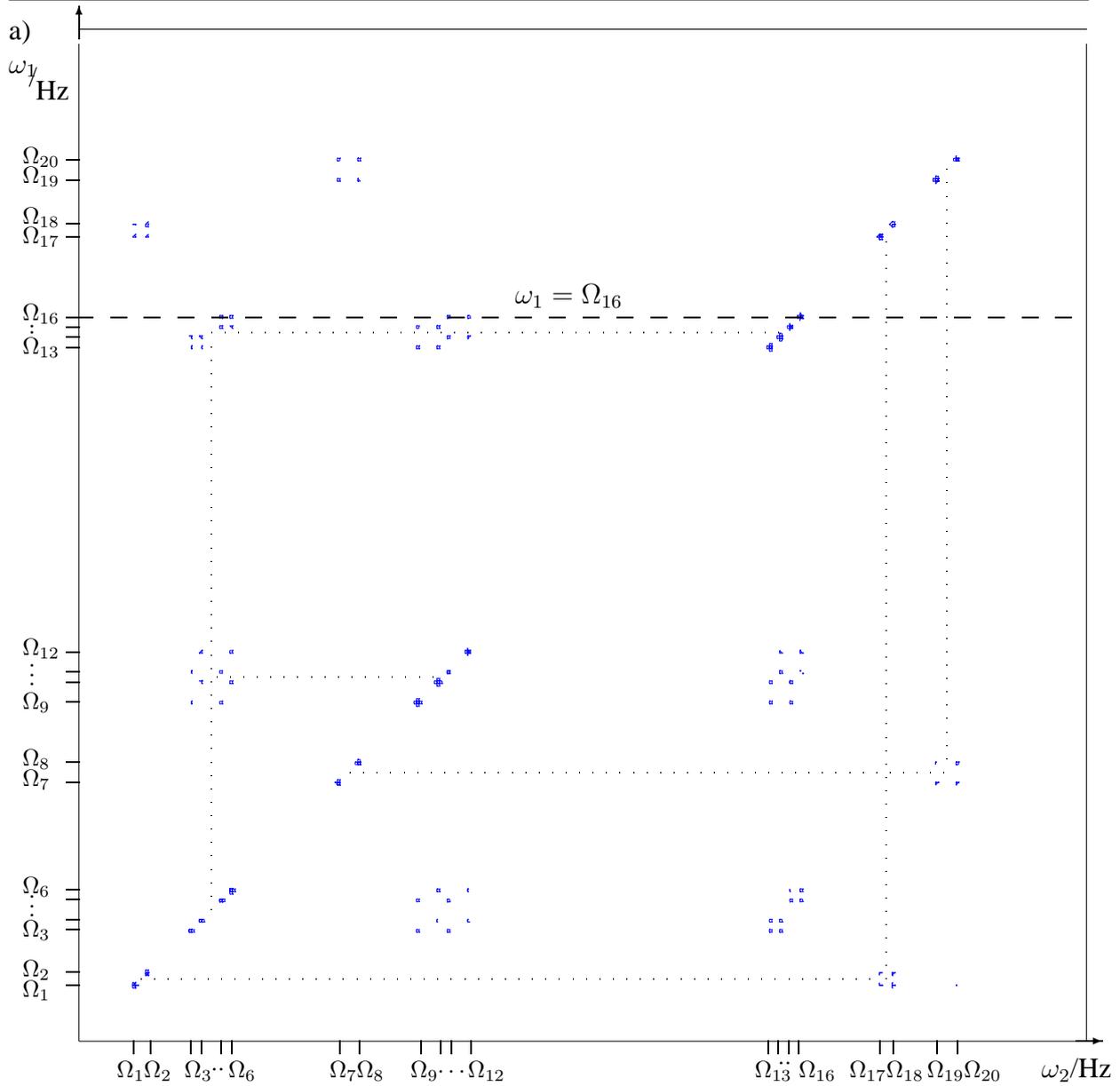


Abb. 4.9: a) 2D-Schnitt $K_3(\omega_1, \omega_2, -\omega_2)$ (Magnitudendarstellung) für ein Beispielsystem aus zwei AX- und einem AMX-Spinsystemen und b) phasenkorrigierter 1D-Schnitt durch das 2D-Spektrum bei $\omega_1 = \Omega_{16}$.

Eine Analyse der Tabelle oder des Spektrums zeigt, dass das in Abbildung (4.8) dargestellte 1D-Spektrum das Ergebnis einer Superposition von drei Spinsystemen ist. In Abbildung (4.9a) sind anhand der gepunkteten Linie zwei AX-Spinsysteme und ein AMX-Spinsystem deutlich erkennbar. Der 2D-Schnitt zeigt auch die für *E.COSY*-Spektren typischen Kreuzsignale für das AMX-Spinsystem. In Abbildung (4.9b) ist ein eindimensionaler Schnitt durch die Frequenzachse ω_1 bei Ω_{16} dargestellt. Neben den in Absorption befindlichen Diagonalsignalen sind die in Antiphase stehenden Kreuzsignale zu erkennen. Die Vorzeichen der Gewichte in der Matrix C entsprechen den im phasenkorrigierten 2D-Spektrum vorkommenden Signalen. Die Absolutbeträge stimmen mit den Spurintensitäten nicht überein, da sich die Signale im 2D-Spektrum aus mehreren benachbarten Spuren zusammensetzen. Sie geben aber einen Anhaltspunkt zur Intensität des jeweiligen Signals.

4.2.2 Trennung von überlagerten Spinsystemen mit überlappenden Signalen

In Abbildung (4.10) ist ein weiteres simuliertes eindimensionales Spektrum eines potenziell unbekanntem Spinsystems (Spinparameter in Anhang A.1.4) mit möglichen Koppelungsschemata gegeben.

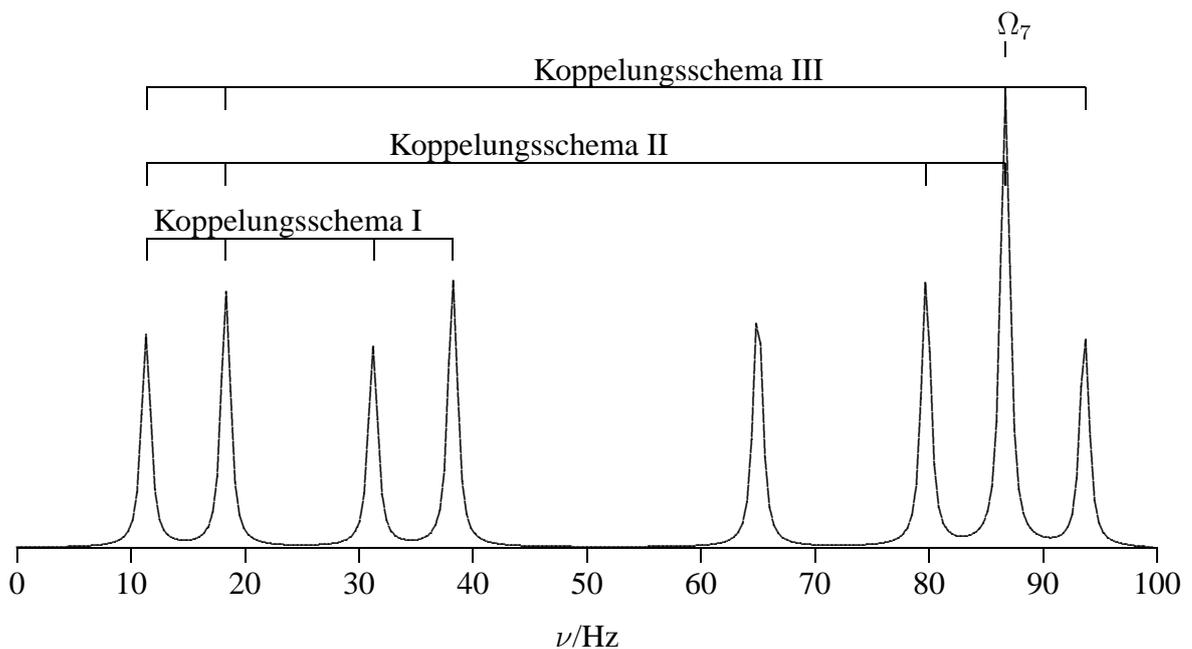


Abb. 4.10: 1D-Spektrum eines unbekanntem Beispielspinsystems. Über den Signalen sind mögliche Koppelungsschemata dargestellt. Die Frequenz bei $\Omega_7 = 86,5$ Hz ist gesondert markiert und könnte das Ergebnis von überlappenden Signalen sein.

Die acht beobachteten Resonanzen werden dem neuronalen Netz als Vorinformation präsentiert. Die resultierenden Werte der Gewichtsmatrix C sind als gerundete Werte in Tabelle (4.2) gegeben. Das über den Volterra-Kern dritter Ordnung berechnete zweidimensionale Magnitudenspektrum ist in Abbildung (4.11) dargestellt.

Frequenz	Ω_1	Ω_2	Ω_3	Ω_4	Ω_5	Ω_6	Ω_7	Ω_8
Ω_1	25,2	0,2	-0,1	-0,0	0,1	-0,5	4,4	-6,8
Ω_2	0,2	25,7	-0,2	0,1	0,5	-0,6	-3,0	6,5
Ω_3	-0,1	-0,2	22,6	-0,3	0,6	5,3	-3,3	-0,9
Ω_4	-0,0	0,1	-0,3	29,1	-0,4	-5,9	5,0	-0,3
Ω_5	0,1	0,5	0,6	-0,4	27,6	0,5	0,6	-0,0
Ω_6	-0,5	-0,6	5,3	-5,9	0,5	28,6	0,7	0,5
Ω_7	4,4	-3,0	-3,3	5,0	0,6	0,7	26,2	0,8
Ω_8	-6,8	6,5	-0,9	-0,3	-0,0	0,5	0,8	23,3

Tabelle 4.2: Gewichtsmatrix C eines unbekanntes Spinsystems mit acht Resonanzfrequenzen als Vorinformation. Bei Ω_7 treten Konnektivitäten zu zwei AX-Systemen auf.

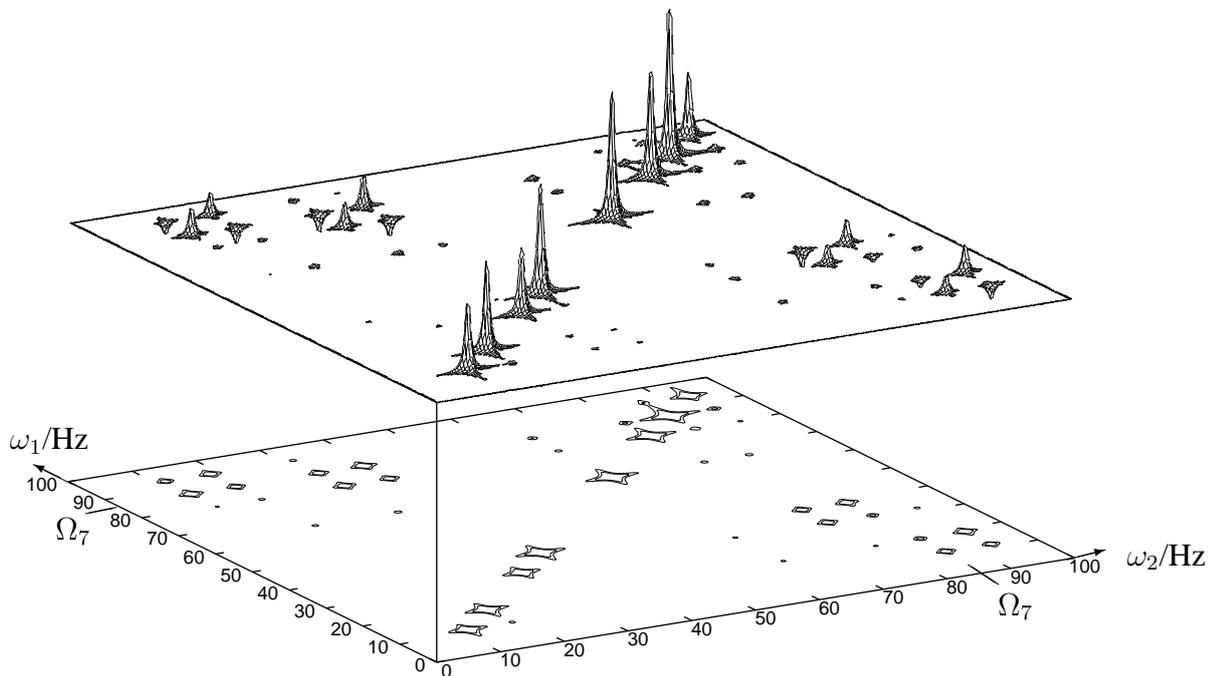


Abb. 4.11: Phasenkorrigierter 2D-Schnitt $K_3(\omega_1, \omega_2, -\omega_2)$ eines unbekanntes Spinsystems mit acht Resonanzfrequenzen als Vorinformation. Bei Ω_7 ist die Superposition von zwei AX-Spinsystemen zu erkennen.

Die Betrachtung von Tabelle (4.2) oder alternativ von Abbildung (4.11) lässt erkennen, dass die Linie bei Ω_7 keine Resonanz eines einzelnen Spins ist. Das System besteht wieder aus einer Superposition von zwei AX-Systemen und einem P-System. Das neuronale Netz ist daher noch einmal mit neun Resonanzen trainiert worden, wobei anstelle der Frequenz Ω_7 leicht unterschiedliche (in der Größenordnung der Linienbreite) Frequenzen Ω_7 und $\Omega_{7'}$ spezifiziert wurden. Das Resultat in Tabelle (4.3) und Abbildung (4.12) zeigt weniger Rauschspuren und eine deutlichere Trennung der Kreuzsignale als bei der Verwendung einer einzelnen Linie bei Ω_7 .

Frequenz	Ω_1	Ω_2	Ω_3	Ω_4	Ω_5	Ω_6	Ω_7	$\Omega_{7'}$	Ω_8
Ω_1	18,3	-0,0	-0,4	0,1	0,8	-0,4	-1,3	7,3	-5,0
Ω_2	-0,0	25,4	0,0	-0,5	0,9	0,4	1,0	-6,9	5,6
Ω_3	-0,4	0,0	18,2	-0,2	-0,3	5,3	-5,0	0,0	0,2
Ω_4	0,1	-0,5	-0,2	23,6	-0,1	-4,2	5,2	0,5	0,6
Ω_5	0,8	0,9	-0,3	-0,1	22,8	-0,1	1,1	-0,8	-0,5
Ω_6	-0,4	0,4	5,3	-4,2	-0,1	22,5	0,9	-1,0	-0,6
Ω_7	-1,3	1,0	-5,0	5,2	1,1	0,9	27,3	-8,4	-0,6
$\Omega_{7'}$	7,3	-6,9	0,0	0,5	-0,8	-1,0	-8,4	33,2	2,1
Ω_8	-5,0	5,6	0,2	0,6	-0,5	-0,6	-0,6	2,1	19,0

Tabelle 4.3: Gewichtsmatrix C für zwei AX-Spinsysteme und einem Singulett mit neun Resonanzfrequenzen als Vorinformation.

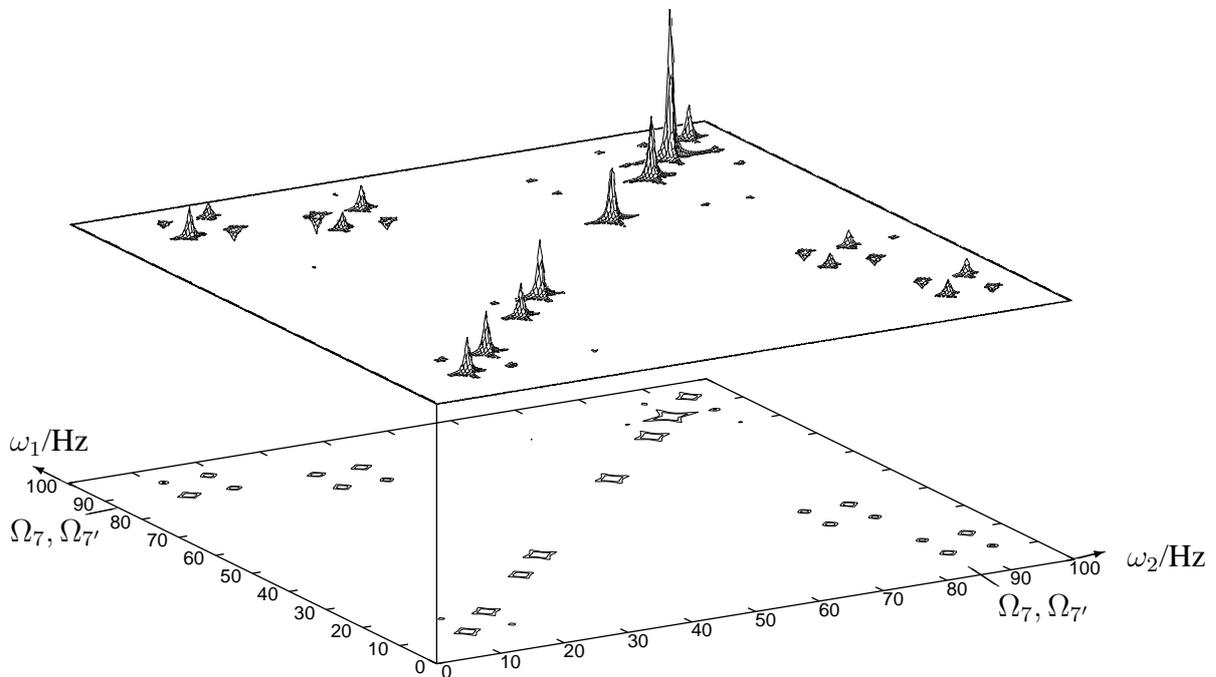


Abb. 4.12: Phasenkorrigierter 2D-Schnitt $K_3(\omega_1, \omega_2, -\omega_2)$ für zwei AX-Spinsysteme und einem Singulett mit neun Resonanzfrequenzen als Vorinformation.

Um die Annahme zu prüfen, dass $\Omega_1, \Omega_2, \Omega_7$ und Ω_8 in dem gemessenen 1D-Spektrum ein AX-System bilden (Koppelungsschema III), wäre in der konventionellen NMR-Spektroskopie eine zweidimensionale Messung eine Lösungsmöglichkeit. Bei Verwendung der neuronalen Netze mussten diesem nur die fraglichen vier Resonanzen als Vorinformation präsentiert werden. Das resultierende Netz besteht aus einer 4×4 -Matrix; der Zusammenhang des Teilspinsystems wird in wenigen Sekunden erlernt. Das Ergebnis ist in Tabelle (4.4) und Abbildung (4.13) dargestellt.

Frequenz	Ω_1	Ω_2	Ω_7	Ω_8
Ω_1	27,0	0,0	4,5	-4,7
Ω_2	0,0	22,5	-2,7	6,4
Ω_7	4,5	-2,7	22,7	-0,3
Ω_8	-4,7	6,4	-0,3	25,6

Tabelle 4.4: Gewichtsmatrix C mit vier Resonanzfrequenzen als Vorinformation zur Prüfung des Koppelungsschemas III.

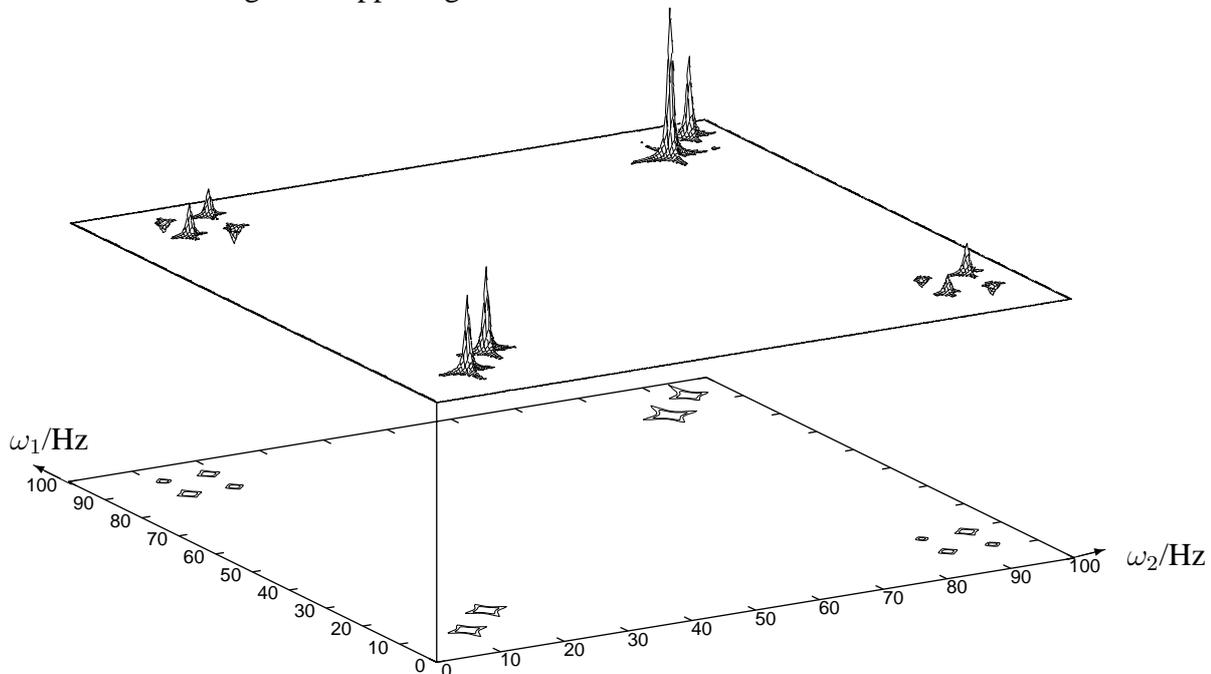


Abb. 4.13: Phasenkorrigierter 2D-Schnitt $K_3(\omega_1, \omega_2, -\omega_2)$ mit vier Resonanzfrequenzen als Vorinformation zur Prüfung des Koppelungsschemas III.

Alle Resonanzen, die dem Netz nicht als Vorinformation gegeben werden, erscheinen nicht im Spektrum. Neben der effektiven Filterung nichtrelevanter Daten ist die Zugehörigkeit des Systems nach dem Koppelungsschema III deutlich zu erkennen.

Wird dem neuronalen Netz ein nicht koppelndes Teilspinsystem präsentiert, so wird kein Zusammenhang erlernt. Bei Vorgabe des Koppelungsschemas I ($\Omega_1, \Omega_2, \Omega_3$ und Ω_4) verschwinden die Nichtdiagonalelemente der Gewichtsmatrix (Tabelle (4.5)) und es erscheinen keine Kreuzsignale im 2D-Spektrum (Abbildung (4.14)).

Frequenz	Ω_1	Ω_2	Ω_3	Ω_4
Ω_1	22,8	1,7	0,0	-0,9
Ω_2	1,7	24,0	-0,1	-0,7
Ω_3	0,0	-0,1	17,9	0,9
Ω_4	-0,9	-0,7	0,9	24,7

Tabelle 4.5: Gewichtsmatrix C mit vier Resonanzfrequenzen als Vorinformation, die nicht demselben koppelnden Spinsystem angehören (Koppelungsschema I).

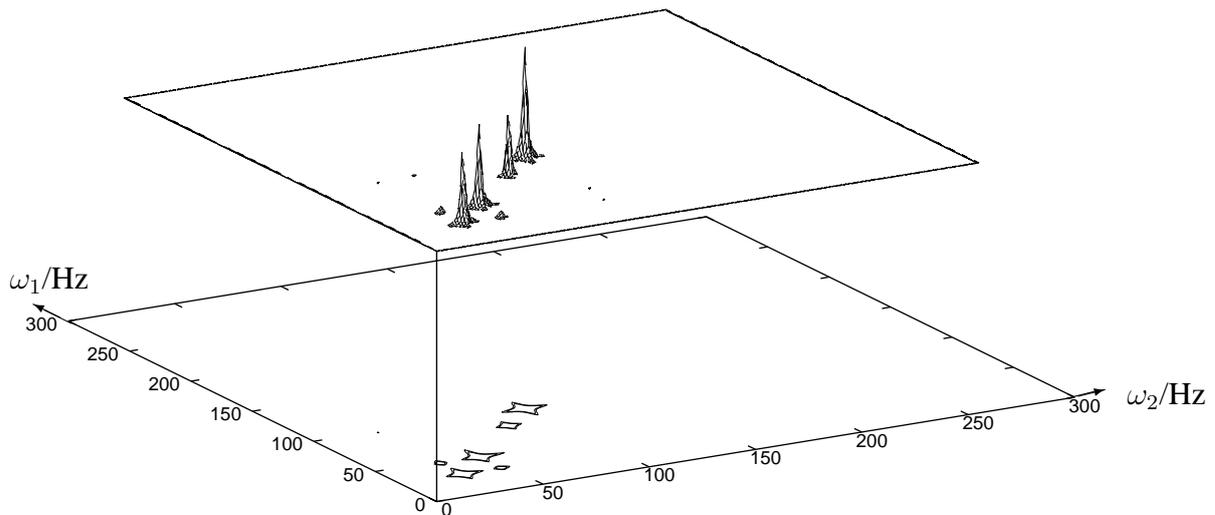


Abb. 4.14: Phasenkorrigierter 2D-Schnitt $K_3(\omega_1, \omega_2, -\omega_2)$ mit vier Resonanzfrequenzen als Vorinformation, die nicht demselben koppelnden Spinsystem angehören (Koppelungsschema I).

4.2.3 Quadraturdetektion

Um eine t_1 -Quadraturdetektion mit dieser Art von Netzen durchführen zu können, wurde die Gewichtsmatrix auf komplexe Größen erweitert. Als Eingabe dienen hier komplexe Zeitfunktionen, die durch Faltung der komplexen Anregungssequenz mit den komplexen Filtern bei Ω_i entstehen. Durch die Positionierung der Trägerfrequenz in die Mitte des Spektrums verdoppelt sich die Abtastzeit und die Gedächtnislänge L wird halbiert. Aus der komplexen Gewichtsmatrix eines trainierten Netzes können vorzeichensensitive Spektren berechnet werden.

In Abbildung (4.15) ist das simulierte 1D-Spektrum von 4-Aminobuttersäure, einem wichtigen Neuroinhibitor, dargestellt. Die von der Aminogruppe erzeugte Resonanzfrequenz bei $\nu = -76$ Hz dient als Kontrollsignal. Die Spinparameter sind im Anhang A.1.5 gegeben.

Mit den zwölf detektierbaren Resonanzfrequenzen ist ein komplexes Netz trainiert worden. Der Realteil der resultierenden komplexen Gewichtsmatrix ist in Tabelle (4.6) dargestellt, der Imaginärteil in Tabelle (4.7).

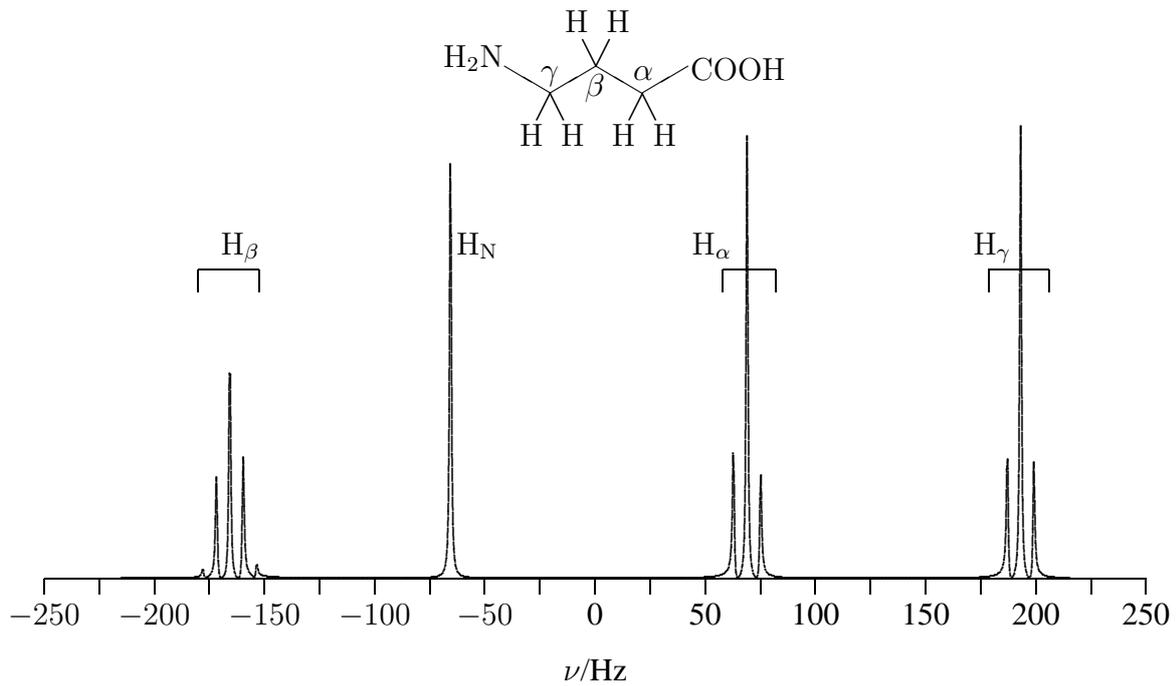


Abb. 4.15: Simuliertes 1D-Spektrum in Quadratur von 4-Aminobuttersäure (*power-Darstellung*). Über dem Spektrum ist die Zugehörigkeit der Signale gekennzeichnet.

Freq.	Ω_1	Ω_2	Ω_3	Ω_4	Ω_5	Ω_6	Ω_7	Ω_8	Ω_9	Ω_{10}	Ω_{11}	Ω_{12}
Ω_1	12,1	-2,1	1,0	-0,2	0,3	-0,3	0,2	0,3	-0,7	0,6	0,5	-1,5
Ω_2	-2,1	12,9	-0,5	0,9	0,2	0,0	1,2	1,2	-0,1	1,9	1,1	-0,2
Ω_3	1,0	-0,5	12,9	-8,3	1,0	0,3	0,9	1,0	0,6	0,6	1,5	0,2
Ω_4	-0,2	0,9	-8,3	12,6	-1,0	0,3	-0,5	1,2	1,2	-0,5	0,6	1,5
Ω_5	0,3	0,2	1,0	-1,0	10,0	0,0	-1,3	0,0	0,5	-1,9	-0,5	1,3
Ω_6	-0,3	0,0	0,3	0,3	0,0	12,8	-0,3	0,1	-0,1	0,0	0,0	-0,1
Ω_7	0,2	1,2	0,9	-0,5	-1,3	-0,3	13,1	-0,4	-0,1	1,3	-0,2	-0,3
Ω_8	0,3	1,2	1,0	1,2	0,0	0,1	-0,4	12,4	-2,2	-0,4	0,4	-0,1
Ω_9	-0,7	-0,1	0,6	1,2	0,5	-0,1	-0,1	-2,2	12,9	-0,3	-0,1	-0,3
Ω_{10}	0,6	1,9	0,6	-0,5	-1,9	0,0	1,3	-0,4	-0,3	8,7	0,4	-0,1
Ω_{11}	0,5	1,1	1,5	0,6	-0,5	0,0	-0,2	0,4	-0,1	0,4	14,3	-1,9
Ω_{12}	-1,5	-0,2	0,2	1,5	1,3	-0,1	-0,3	-0,1	-0,3	-0,1	-1,9	14,9

Tabelle 4.6: Realteil der komplexen Gewichtsmatrix **C** für 4-Aminobuttersäure mit zwölf Resonanzfrequenzen als Vorinformation.

Freq.	Ω_1	Ω_2	Ω_3	Ω_4	Ω_5	Ω_6	Ω_7	Ω_8	Ω_9	Ω_{10}	Ω_{11}	Ω_{12}
Ω_1	-13,4	2,3	-1,0	0,6	-0,5	0,0	-0,6	-0,5	1,2	-0,5	-0,1	0,7
Ω_2	2,3	-15,0	0,5	-1,4	0,0	-0,1	-0,7	-1,2	0,1	-0,8	-1,2	0,0
Ω_3	-1,0	0,5	-12,9	8,1	-0,8	-0,2	-1,0	-0,8	-0,6	-1,2	-1,0	-0,2
Ω_4	0,6	-1,4	8,1	-12,4	0,9	-0,3	0,4	-1,2	-1,1	0,1	-1,4	-1,5
Ω_5	-0,5	0,0	-0,8	0,9	-10,1	0,1	1,7	0,2	-0,9	0,9	1,3	-1,3
Ω_6	0,0	-0,1	-0,2	-0,3	0,1	-12,0	0,0	-0,1	0,1	0,3	0,2	0,4
Ω_7	-0,6	-0,7	-1,0	0,4	1,7	0,0	-13,0	0,2	0,5	-0,1	1,2	0,2
Ω_8	-0,5	-1,2	-0,8	-1,2	0,2	-0,1	0,2	-11,0	0,9	0,1	-0,4	0,1
Ω_9	1,2	0,1	-0,6	-1,1	-0,9	0,1	0,5	0,9	-13,0	0,5	0,2	0,2
Ω_{10}	-0,5	-0,8	-1,2	0,1	0,9	0,3	-0,1	0,1	0,5	-10,6	-12,1	1,2
Ω_{11}	-0,1	-1,2	-1,0	-1,4	1,3	0,2	1,2	-0,4	0,2	-12,1	-12,1	0,2
Ω_{12}	0,7	0,0	-0,2	-1,5	-1,3	0,4	0,2	0,1	0,2	1,2	0,2	-14,4

Tabelle 4.7: Imaginärteil der komplexen Gewichtsmatrix C für 4-Aminobuttersäure mit zwölf Resonanzfrequenzen als Vorinformation.

Auffällig sind die alternierenden Vorzeichen in beiden Tabellen. Durch die hervorgehobenen Werte können die Konnektivitäten zwischen den Protonen H_α und H_β , sowie zwischen H_β und H_γ ermittelt werden. Zwischen H_α und H_γ , sowie für das Signal der Amidprotonen sind erwartungsgemäß keine Konnektivitäten ersichtlich. Dieses wird auch durch die Magnitudendarstellung der Matrix in Tabelle (4.8) gezeigt.

Freq.	Ω_1	Ω_2	Ω_3	Ω_4	Ω_5	Ω_6	Ω_7	Ω_8	Ω_9	Ω_{10}	Ω_{11}	Ω_{12}
Ω_1	18,1	3,1	1,4	0,6	0,6	0,3	0,6	0,5	1,4	0,8	0,5	1,7
Ω_2	3,1	19,7	0,7	1,7	0,2	0,1	1,4	1,6	0,2	2,1	1,6	0,2
Ω_3	1,4	0,7	18,2	11,5	1,3	0,4	1,3	1,4	0,8	1,3	1,8	0,3
Ω_4	0,6	1,7	11,5	17,7	1,3	0,4	0,6	1,7	1,6	0,5	1,5	2,1
Ω_5	0,6	0,2	1,3	1,3	14,2	0,1	2,2	0,2	1,0	2,1	1,4	1,9
Ω_6	0,3	0,1	0,4	0,4	0,1	17,5	0,3	0,2	0,1	0,4	0,3	0,4
Ω_7	0,6	1,4	1,3	0,6	2,2	0,3	18,5	0,5	0,5	1,3	1,2	0,4
Ω_8	0,5	1,6	1,4	1,7	0,2	0,2	0,5	16,5	2,4	0,4	0,6	0,1
Ω_9	1,4	0,2	0,8	1,6	1,0	0,1	0,5	2,4	18,3	0,5	0,2	0,3
Ω_{10}	0,8	2,1	1,3	0,5	2,1	0,4	1,3	0,4	0,5	13,7	12,1	1,2
Ω_{11}	0,5	1,6	1,8	1,5	1,4	0,3	1,2	0,6	0,2	12,1	18,7	1,9
Ω_{12}	1,7	0,2	0,3	2,1	1,9	0,4	0,4	0,1	0,3	1,2	1,9	20,7

Tabelle 4.8: Magnitude der komplexen Gewichtsmatrix C für 4-Aminobuttersäure mit zwölf Resonanzfrequenzen als Vorinformation.

Die Tabellenwerte und das zugehörige 2D-Spektrum (Abbildung (4.16)) zeigen, dass das Netz stärkere Konnektivitäten zwischen H_β und H_γ approximiert hat. Einen Grund könnten vorhan-

dene Resonanzen sein, die dem Netz jedoch wegen der fehlenden Auflösung im 1D-Spektrum nicht als Vorinformation gegeben werden konnten. Die Multipllettstruktur der Kreuzsignale ist deutlich sichtbar.

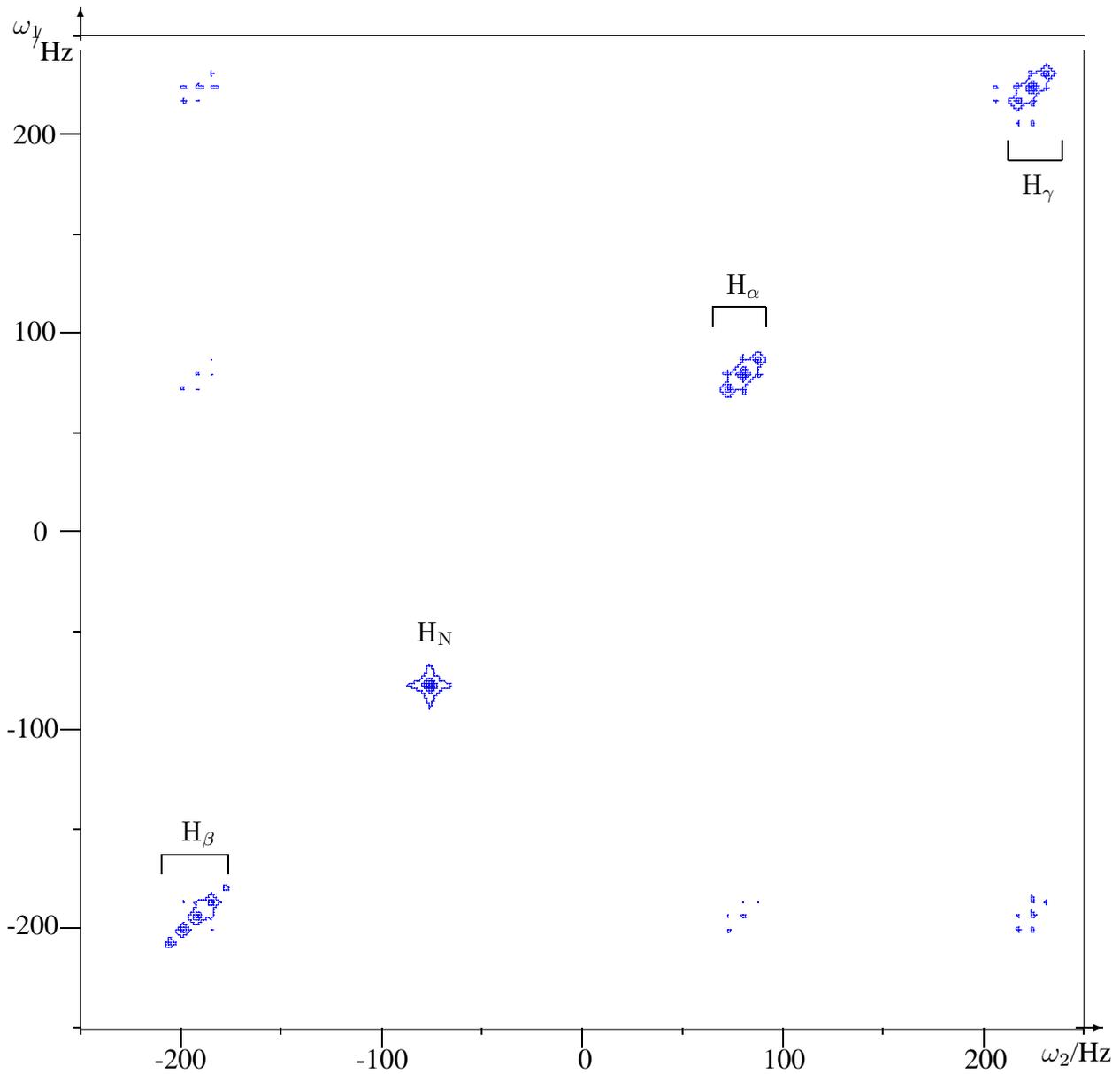


Abb. 4.16: Phasensensitiver 2D-Schnitt $K_3(\omega_1, \omega_2, -\omega_2)$ (Magnitudendarstellung) von 4-Aminobuttersäure. Die Zugehörigkeiten der Protonen ist an den Diagonalsignalen gekennzeichnet.

4.2.4 Spinsysteme höherer Ordnung

Als Beispiel eines Spinsystems höherer Ordnung sei das AA'BB'C Spinsystem des Phenols vorgestellt. Das für eine Spektrometerfrequenz von 400 MHz simulierte 1D-Spektrum mit seinen 37 detektierbaren Signalen ist in Abbildung (4.17) dargestellt (Spinparameter in Anhang A.1.6). Die bei $\nu = 2800$ Hz erscheinende Resonanzfrequenz dient als zusätzliches Kontrollsignal.

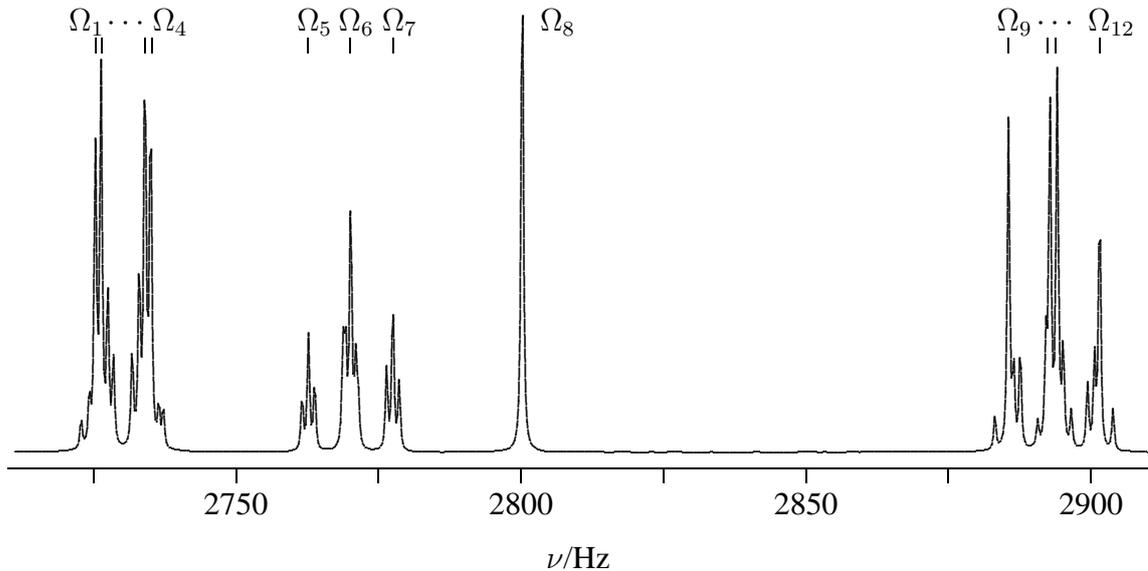


Abb. 4.17: Simuliertes 1D-Spektrum von Phenol. Auserwählte Resonanzfrequenzen sind mit Ω_i markiert.

Die resultierende Gewichtsmatrix mit 38 detektierbaren Resonanzen zeigte schon nach wenigen Lernzyklen an mehreren Stellen verschwindende oder nur geringe Intensität bei den Diagonalelementen. Auch nach Abschluss des Trainings konnte keine Änderung des Sachverhalts beobachtet werden. Das deutet darauf hin, dass bei diesen Frequenzen evtl. Koppelungen nicht aufgelöst wurden. Es wurde daher ein neues Training mit den zwölf Resonanzfrequenzen durchgeführt, die die betragsmäßig stärksten Diagonalelemente aufwiesen und in Abbildung (4.17) auch die dominanten Signale darstellen. Das Ergebnis des Trainings zeigt Tabelle (4.9).

Die relativen Intensitäten der Diagonalelemente werden nicht wiedergegeben, jedoch werden Koppelungen zweifelsfrei erkannt und Nichtkoppelungen ausgeschlossen. Durch die Auswahl von elf Resonanzfrequenzen des Phenols – es sind insgesamt 88 vorhanden – fehlen dem Netz Informationen bei der Approximation des Funktional. Daher kann keine eindeutige Zuordnung erfolgen. Die Annahme des koppelnden Spinsystems kann jedoch durch das in Abbildung (4.18) dargestellte 2D-Spektrum bestätigt werden. Hier zeigt sich der Einfluss der Gewichtskoeffizienten v_i des Ausgabeneurons, die für Spinsysteme höherer Ordnung eine bessere Auswertung eines 2D-Spektrums ermöglichen.

Freq.	Ω_1	Ω_2	Ω_3	Ω_4	Ω_5	Ω_6	Ω_7	Ω_8	Ω_9	Ω_{10}	Ω_{11}	Ω_{12}
Ω_1	14,0	-12,3	5,3	-2,8	-2,2	3,0	-1,9	-0,6	1,6	2,9	-1,7	2,3
Ω_2	-12,3	16,3	-3,4	3,3	2,7	-1,7	2,0	0,6	0,1	-0,8	3,4	-2,0
Ω_3	5,3	-3,4	6,1	-4,4	9,0	-6,7	7,7	-0,4	-2,4	-2,8	8,6	-7,0
Ω_4	-2,8	3,3	-4,4	7,5	-6,5	9,2	-6,8	0,4	2,0	3,2	-5,1	8,4
Ω_5	-2,2	2,7	9,0	-6,5	1,9	-0,8	0,5	0,2	-2,0	4,5	-2,1	0,6
Ω_6	3,0	-1,7	-6,7	9,2	-0,8	3,8	-1,9	-0,2	-0,2	-3,4	6,4	-4,0
Ω_7	-1,9	2,0	7,7	-6,8	0,5	-1,9	1,9	-0,5	-0,3	3,5	-3,7	6,3
Ω_8	-0,6	0,6	-0,4	0,4	0,2	-0,2	-0,5	11,6	-0,5	-0,1	0,4	-0,0
Ω_9	1,6	0,1	-2,4	2,0	-2,0	-0,2	-0,3	-0,5	10,2	-3,1	1,8	-1,4
Ω_{10}	2,9	-0,8	-2,8	3,2	4,5	-3,4	3,5	-0,1	-3,1	10,4	-3,0	-1,0
Ω_{11}	-1,7	3,4	8,6	-5,1	-2,1	6,4	-3,7	0,4	1,8	-3,0	5,7	-3,1
Ω_{12}	2,3	-2,0	-7,0	8,4	0,6	-4,0	6,3	-0,0	-1,4	-1,0	-3,1	9,0

Tabelle 4.9: Gewichtsmatrix C für Phenol und einem Kontrollsignal mit zwölf auserwählten Resonanzfrequenzen als Vorinformation.

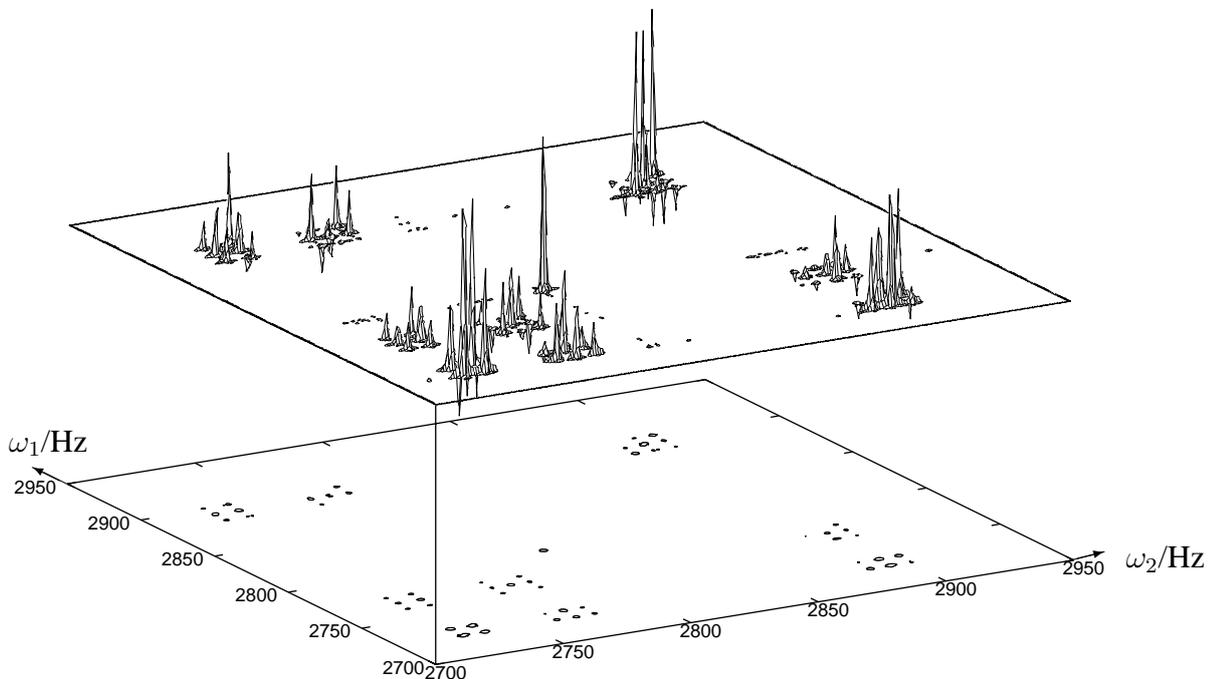


Abb. 4.18: Phasenkorrigierter 2D-Schnitt $K_3(\omega_1, \omega_2, -\omega_2)$ für Phenol und einem Kontrollsignal mit zwölf auserwählten Resonanzfrequenzen als Vorinformation.

4.2.5 Auswahlkriterien für bekannte Summenformeln

In der üblichen NMR-Spektroskopie wird ausgehend vom Spektrum auf eine Zielstruktur hingearbeitet. Dabei kann es vorkommen, dass eine existente Molekülstruktur nicht erkannt wird. Ein neuer Ansatz berechnet bei Bekanntheit der Bruttoformel mit Hilfe eines Strukturgenerators alle möglichen Strukturen zu dieser Bruttoformel. Nach dem Ausschlussprinzip werden dann die in Frage kommenden Strukturen näher untersucht.

Ein neues Kriterium zur Strukturauswahl liefert die *NN*-NMR. Mit Hilfe der neuronalen Anregung können für alle Isomere die Systemantworten berechnet, gespeichert und mit der Antwort der Messprobe nach Gleichung (2.51) korreliert werden. Anhand der Korrelationskoeffizienten kann eine Auswahl stattfinden, die ein Training des neuronalen Netzes und die Ermittlung eines 2D-Spektrums entbehrlich macht.

Als Beispiel seien Verbindungen der Summenformel C_4H_6 betrachtet. In Abbildung (4.19) sind alle möglichen Strukturen dieser Verbindungen dargestellt.

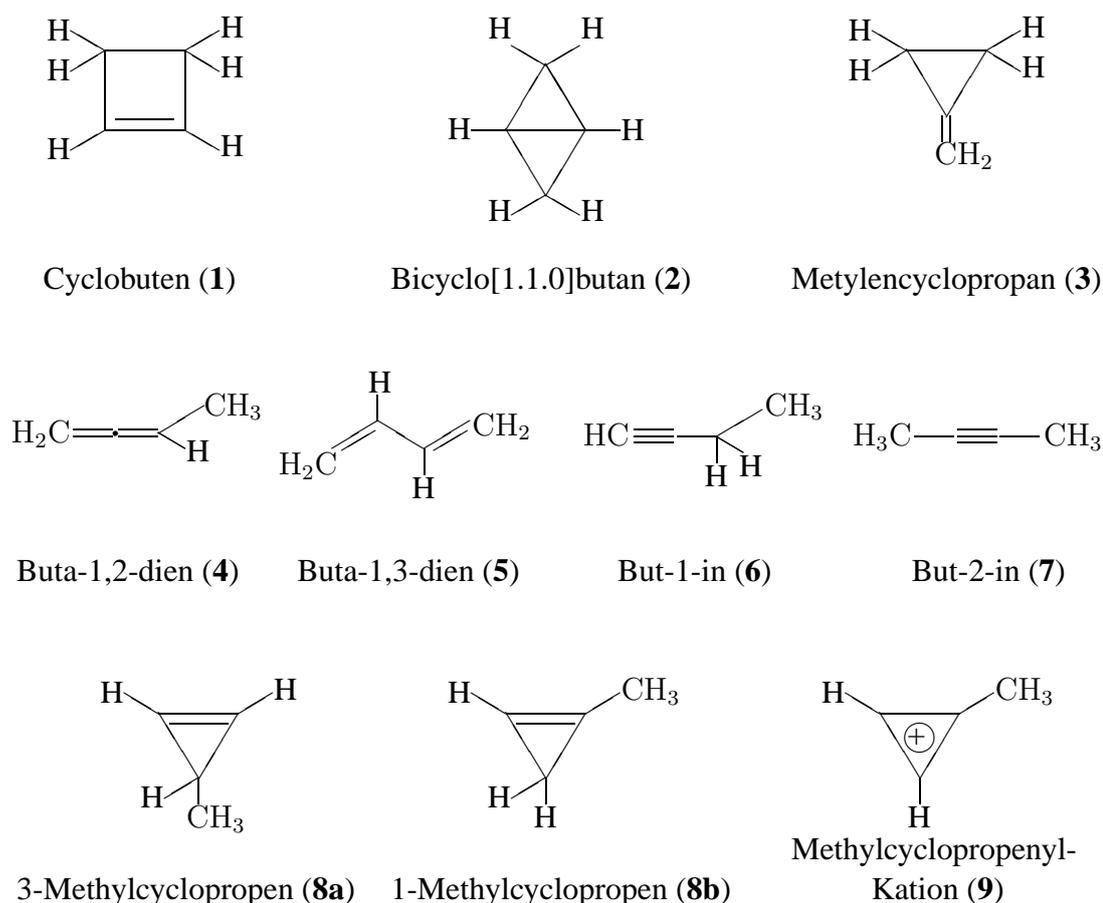


Abb. 4.19: Strukturformeln von Verbindungen mit der Summenformel C_4H_6 . Verbindung (9) hat die Summenformel $C_4H_5^+$ und ist das Produkt aus (8a) oder (8b).

Die beiden Isomere (8a) und (8b) sind nicht stabil und ionisieren zum 2π -Aromaten (9). Somit kommen nur sieben Verbindungen in Betracht, deren NMR-Daten in Anhang A.1.7 angegeben

sind.

Der Einfluss von Messrauschen sowie leichte Variationen der Spinparameter werden in der Praxis Korrelationskoeffizienten von 1 verhindern. Um diesen Einfluss abzuschätzen, sind die Systemdaten leicht zu modifizieren. Die Modifikationen sind mit der Verbindung But-1-in (6) durchgeführt worden. Eine mögliche Größenordnung für die Abweichungen der chemischen Verschiebung kann abgeschätzt werden, indem die Resonanzen zufällig um einige Hertz verändert werden. An dieser Stelle sei erwähnt, dass eine Korrelation von reinen 1D-Spektren schon bei kleinen Abweichungen der chemischen Verschiebung im Promillebereich verschwindend kleine Werte zeigt. Für einen sinnvollen Einsatz sind reine 1D-Spektren daher unbrauchbar.

Die Korrelationen der verschiedenen Systemantworten mit variiertem chemischer Verschiebung von (6) als Funktion der Anzahl der Messpunkte sind in Abbildung (4.20) für eine Spektromerfrequenz von 400 MHz dargestellt. Die abfallenden Anfangswerte in dem Diagramm unterliegen statistischen Ungenauigkeiten und finden daher keine Beachtung.

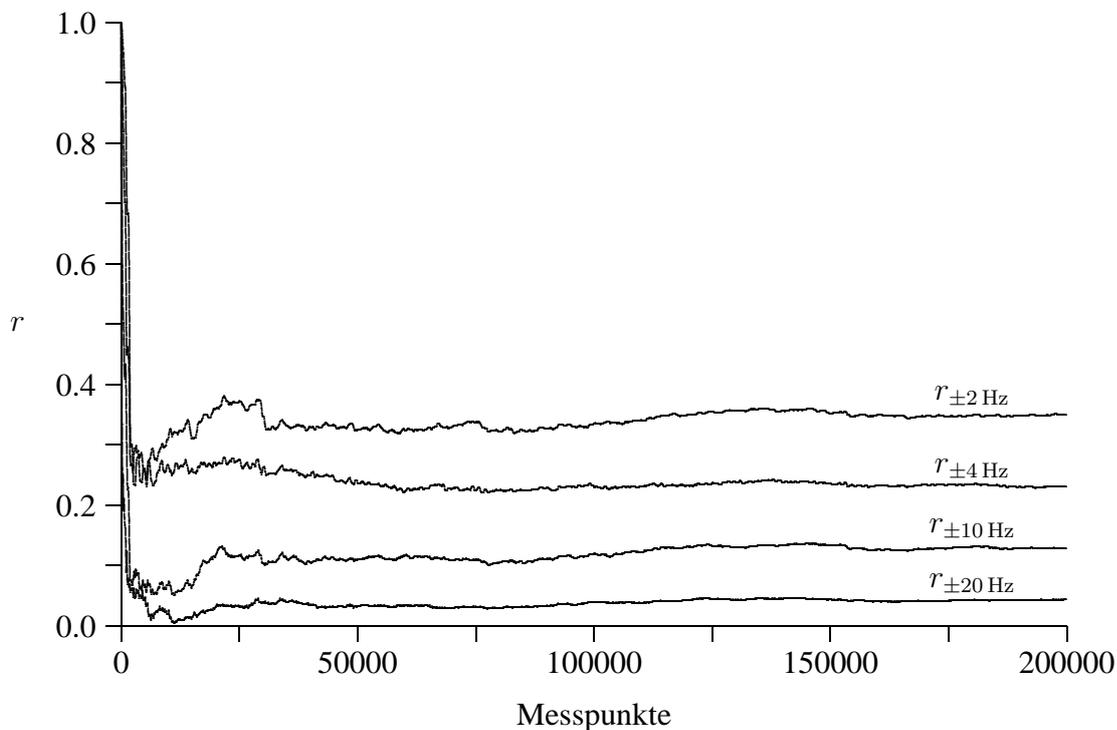


Abb. 4.20: Korrelationskoeffizienten für die Systemantworten des But-1-ins. Die chemischen Verschiebungen wurden zufällig um den im Diagramm angegebenen Wert verändert.

Schon bei einer Variation von ± 2 Hz sinken in der Darstellung die Korrelationskoeffizienten unter 0,4. Wird der 1D-Anteil aus der Systemantwort nach Gleichung (2.52) entfernt, so können deutlich bessere Resultate erzielt werden (Abbildung (4.21)). Die chemischen Verschiebungen der Kerne können bei Berücksichtigung des linearen Anteils bis 0,05 ppm variieren. Das Verfahren ist relativ sensitiv, da Abweichungen der Koppelungskonstanten von bis zu ± 2 Hz keinen signifikanten Einfluss auf die Korrelationen haben. Auch ein zusätzlicher Rauschanteil von 5% kann vernachlässigt werden.

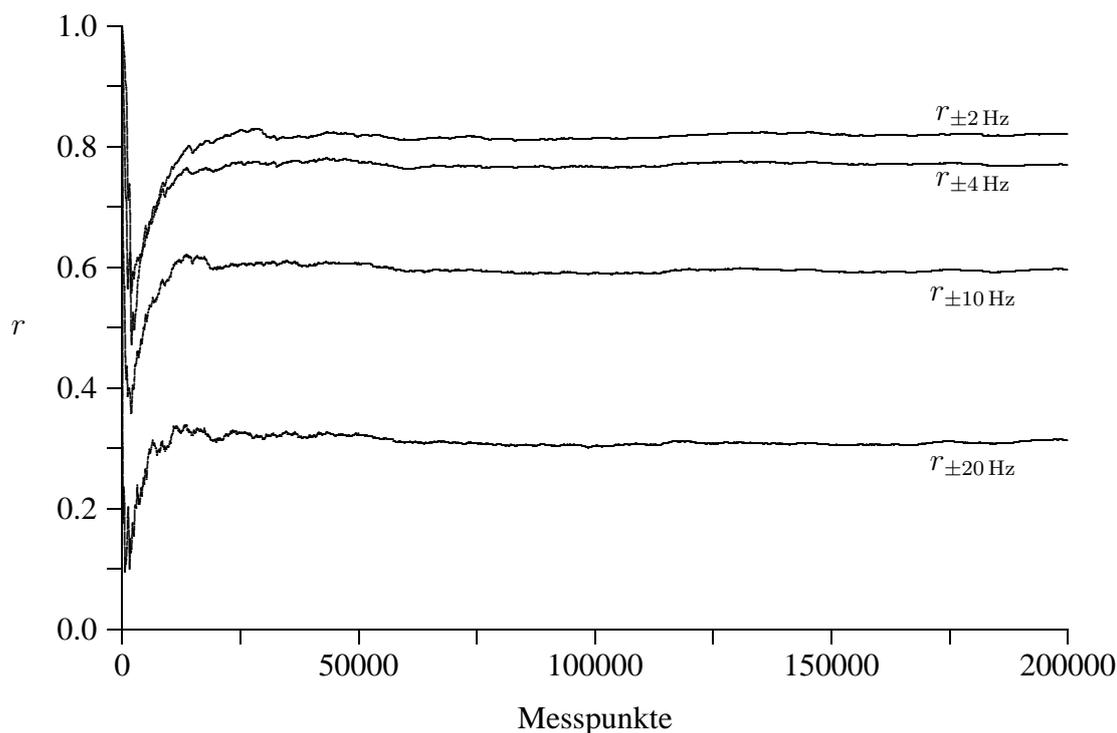


Abb. 4.21: Korrelationskoeffizienten ohne 1D-Anteil für die Systemantworten des But-1-*ins*. Die chemischen Verschiebungen wurden zufällig um den im Diagramm angegebenen Wert verändert.

Um einen Entscheidungsprozess für die Verbindungen der Summenformel C_4H_6 mit diesem Korrelationsverfahren zu testen, sind die Systemantworten aller Isomere simuliert worden. Diese Antworten sind dann mit der simulierten Antwort für **(6)** unter Berücksichtigung des 1D-Anteils korreliert worden. Als „unbekannte“ Substanz wurde **(6)** mit einer Variation der chemischen Verschiebung von ± 4 Hz versehen. Die in Abbildung (4.22) dargestellten Koeffizienten bestätigen schon nach wenigen tausend Messpunkten, dass es sich bei der unbekannt Substanz um **(6)** handelt. Bei der hier angewandten Abtastzeit von $\Delta t = 0,2$ ms entsprechen 10000 Messpunkte einer Messzeit von 2 s.

Die Auswahl des Korrelationsverfahrens kann durch zusätzliche Spektren von trainierten Netzen bekräftigt werden. In Abbildung (4.23) ist das via Volterra-Kern dritter Ordnung berechnete 2D-Teilspektrum von **(6)** dargestellt. Die Besonderheit ist hier, dass es in akzeptabler Rechenzeit trotz der für 400 MHz entsprechend großen spektralen Breite von 2500 Hz möglich war, ein Netz zu trainieren. Durch die Modulation mit den 14 vorgegebenen Frequenzen besitzt die Gewichtsmatrix C die Größe 14×14 und ist von der Gedächtnislänge $L = 20.000$ Datenpunkte unabhängig.

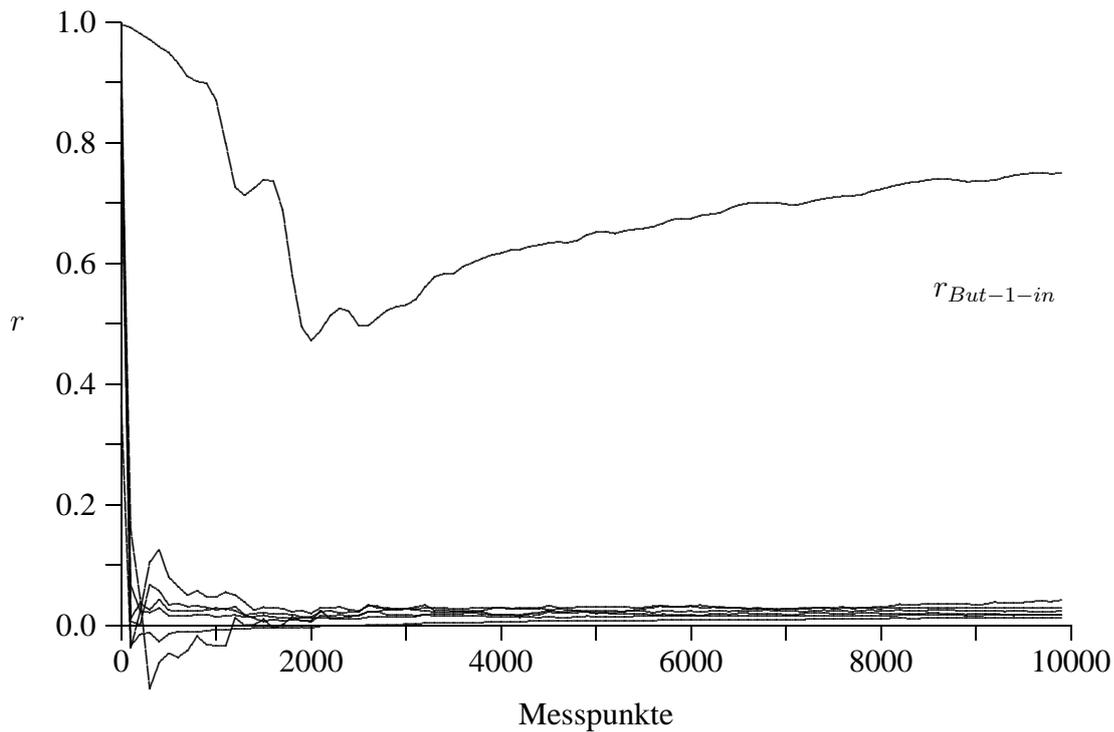


Abb. 4.22: Korrelationskoeffizienten ohne 1D-Anteil für Verbindungen mit der Summenformel C_4H_6 . Bis auf die Korrelationen von But-1-in sind alle anderen verschwindend gering und daher nicht beschriftet. 10000 Messpunkte in diesem Diagramm entsprechen einer Messzeit von 2s.

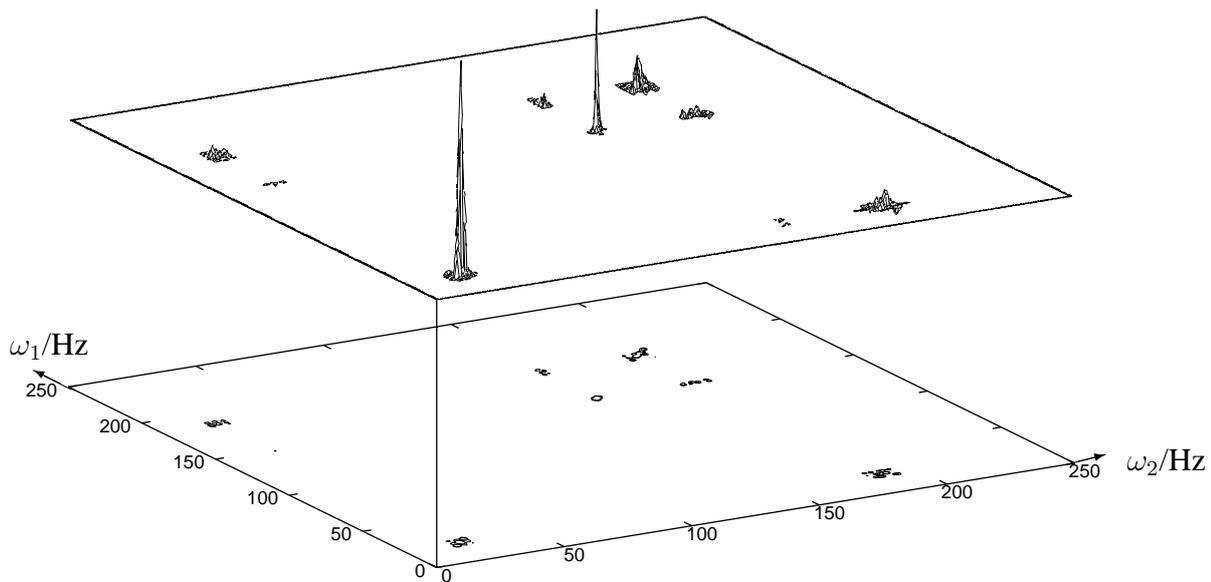


Abb. 4.23: Phasenkorrigierter 2D-Schnitt $K_3(\omega_1, \omega_2, -\omega_2)$ von But-1-in mit 14 Resonanzfrequenzen als Vorinformation. Es ist nur der signifikante Bereich dargestellt.

But-1-in bildet ein A_3M_2X -Spinsystem. In der zugehörigen Tabelle (4.10) sind die eindeutigen Zuordnungen von A ($\Omega_1 - \Omega_3$) mit M ($\Omega_7 - \Omega_{14}$), sowie M mit X ($\Omega_4 - \Omega_6$) gekennzeichnet. Durch die schwache, nicht aufgelöste Koppelung J_{AX} kann wie bei Phenol keine vollständige Zuordnung der Gewichtswerte erfolgen, da dem neuronalen Netz hier zusätzliche Resonanzfrequenzen als Vorinformation fehlen.

Freq.	Ω_1	Ω_2	Ω_3	Ω_4	Ω_5	Ω_6	Ω_7	Ω_8	Ω_9	Ω_{10}	Ω_{11}	Ω_{12}	Ω_{13}	Ω_{14}
Ω_1	16,2	0,1	0,6	0,3	-0,2	0,5	0,5	1,6	2,3	2,0	-0,8	0,4	-2,3	-1,6
Ω_2	0,1	18,9	0,2	0,0	-0,8	-0,1	0,1	-0,1	0,4	0,6	0,3	1,3	-0,1	-0,1
Ω_3	0,6	0,2	17,3	-0,1	-0,5	0,3	-0,9	-0,7	-1,8	0,6	0,7	0,9	0,4	0,8
Ω_4	0,3	0,0	-0,1	17,8	0,6	0,6	2,5	-2,0	1,7	-0,3	1,5	-0,3	0,7	-0,7
Ω_5	-0,2	-0,8	-0,5	0,6	15,5	-0,9	-0,0	-0,0	-0,5	0,4	0,9	-0,3	-0,1	-0,5
Ω_6	0,5	-0,1	0,3	0,6	-0,9	19,7	0,1	0,4	0,0	1,0	-0,8	-0,0	-0,9	-0,7
Ω_7	0,5	0,1	-0,9	2,5	-0,0	0,1	9,3	0,5	-0,8	1,4	1,4	-1,6	0,3	-4,4
Ω_8	1,6	-0,1	-0,7	-2,0	-0,0	0,4	0,5	9,1	1,3	-1,4	-0,6	1,5	-2,7	-1,5
Ω_9	2,3	0,4	-1,8	1,7	-0,5	0,0	-0,8	1,3	14,2	-0,6	-0,6	0,7	0,5	-1,1
Ω_{10}	2,0	0,6	0,6	-0,3	0,4	1,0	1,4	-1,4	-0,6	10,9	0,6	-7,9	0,5	-0,3
Ω_{11}	-0,8	0,3	0,7	1,5	0,9	-0,8	1,4	-0,6	-0,6	0,6	18,9	-0,4	0,7	-0,1
Ω_{12}	0,4	1,3	0,9	-0,3	-0,3	-0,0	-1,6	1,5	0,7	-7,9	-0,4	8,7	-0,2	0,3
Ω_{13}	-2,3	-0,1	0,4	0,7	-0,1	-0,9	0,3	-2,7	0,5	0,5	0,7	-0,2	11,8	1,7
Ω_{14}	-1,6	-0,1	0,8	-0,7	-0,5	-0,7	-4,4	-1,5	-1,1	-0,3	-0,1	0,3	1,7	8,6

Tabelle 4.10: Gewichtsmatrix C für But-1-in mit 14 Resonanzfrequenzen als Vorinformation.

4.2.6 Konnektivitätsbestimmung aus experimentellen Daten

Um die Ergebnisse der Simulationen experimentell zu bestätigen, ist sowohl das konventionelle 1D-Spektrum (Abbildung (4.24)) als auch die Systemantwort auf eine neuronale Anregung des 1-Chlor-2,4-dinitrobenzols gemessen worden.

Die acht Resonanzfrequenzen der zu detektierenden Signale – in Abbildung (4.24) markiert – wurden als Vorinformation einem neuronalen Netz zusammen mit der Systemantwort zum Training präsentiert. Die resultierende Gewichtsmatrix C weist mehrfach erkennbare Konnektivitäten, jedoch keine Multipletstrukturen auf. Aus dem in Abbildung (4.25) dargestellten 2D-Spektrum ist ein Koppelungsschema A- zum X-Kern ersichtlich. Mit acht Resonanzen wurde dem Netz wieder ein zu geringer Informationsgehalt vorgegeben. Das 2D-Spektrum ist auswertbar, nicht jedoch die Matrix C. Um hierfür bessere Resultate zu erzielen, wurde ein zweites Training mit zwölf Resonanzen als Vorinformation durchgeführt, wobei die Resonanzen für den A- und X-Kern ($\Omega_1, \Omega_2, \Omega_7$ und Ω_8) um jeweils eine Linienbreite verdoppelt wurden.

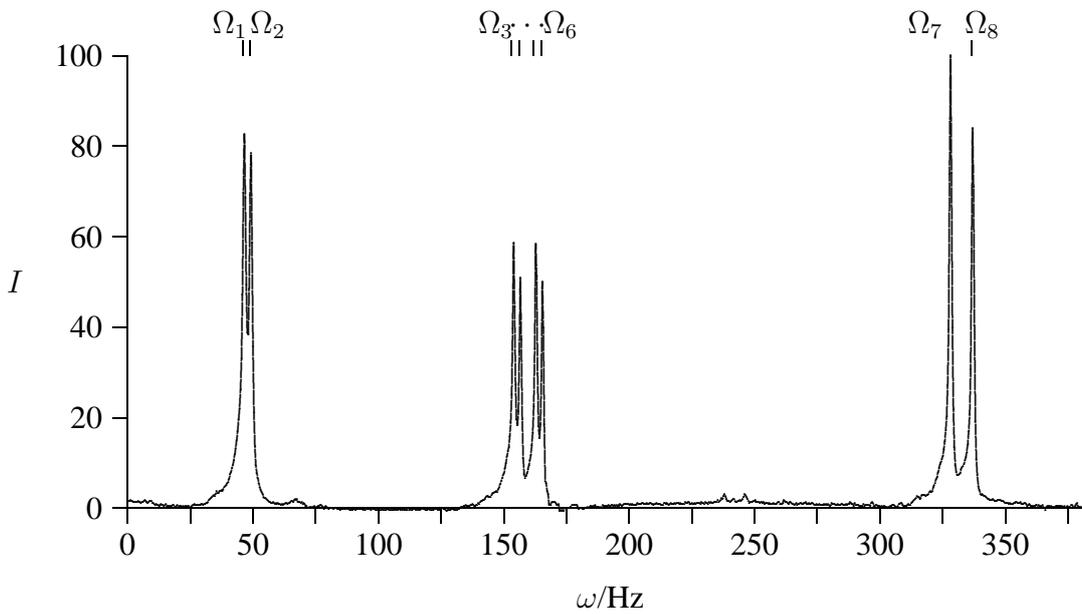


Abb. 4.24: Konventionell gemessenes, phasenkorrigiertes 1D-Spektrum von 1-Chlor-2,4-dinitrobenzol. Die detektierbaren Signale sind mit Ω_i markiert.

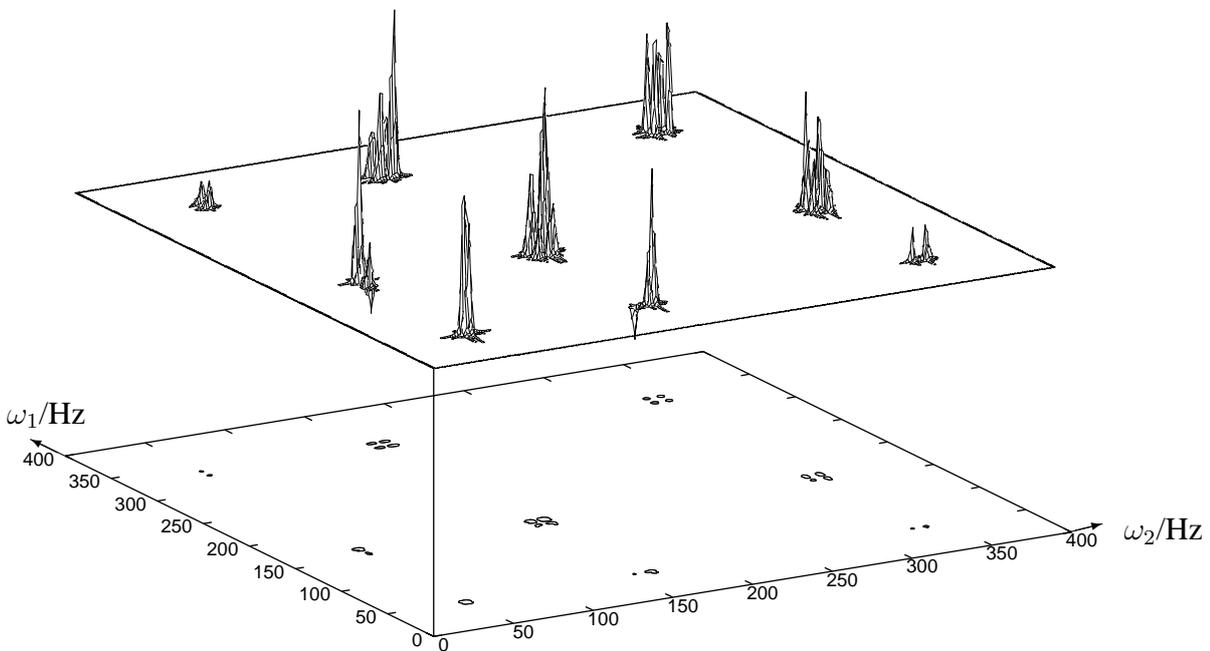


Abb. 4.25: Phasenkorrigierter 2D-Schnitt $K_3(\omega_1, \omega_2, -\omega_2)$ von 1-Chlor-2,4-dinitrobenzol mit acht Resonanzfrequenzen als Vorinformation.

Die in Tabelle (4.11) dargestellten Werte zeigen das Verhalten für das AMX-Spinsystem. Alle Konnektivitäten können zugeordnet werden. Bei den Konnektivitäten zwischen dem A- und X-Kern sind die Beträge sehr gering, jedoch sind die alternierenden Vorzeichen zu erkennen.

Freq.	Ω_1	$\Omega_{1'}$	Ω_2	$\Omega_{2'}$	Ω_3	Ω_4	Ω_5	Ω_6	Ω_7	$\Omega_{7'}$	Ω_8	$\Omega_{8'}$
Ω_1	11,0	-0,6	-0,2	-0,3	0,6	-0,8	0,1	0,0	0,0	-0,0	-0,1	0,1
$\Omega_{1'}$	-0,6	10,1	0,3	0,3	0,3	0,3	1,8	-1,5	0,0	0,1	0,2	-0,1
Ω_2	-0,2	0,3	10,1	0,1	-1,1	0,8	-0,1	-0,1	0,2	-0,1	0,0	0,1
$\Omega_{2'}$	-0,3	0,3	0,1	10,1	0,5	0,7	-0,8	0,9	-0,2	0,2	0,0	0,1
Ω_3	0,6	0,3	-1,1	0,5	9,9	0,3	-0,2	-0,2	2,0	0,4	-1,1	0,2
Ω_4	-0,8	0,3	0,8	0,7	0,3	9,6	-0,3	-0,5	0,2	0,9	0,2	-0,9
Ω_5	0,1	1,8	-0,1	-0,8	-0,2	-0,3	9,5	-0,4	-0,9	-0,1	1,1	-0,2
Ω_6	0,0	-1,5	-0,1	0,9	-0,2	-0,5	-0,4	9,8	-0,5	-1,4	-0,3	1,2
Ω_7	0,0	0,0	0,2	-0,2	2,0	0,2	-0,9	-0,5	10,7	-0,2	-0,3	0,0
$\Omega_{7'}$	-0,0	0,1	-0,1	0,2	0,4	0,9	-0,1	-1,4	-0,2	11,1	-0,1	0,1
Ω_8	-0,1	0,2	0,0	0,0	-1,1	0,2	1,1	-0,3	-0,3	-0,1	11,8	-1,7
$\Omega_{8'}$	0,1	-0,1	0,1	0,1	0,2	-0,9	-0,2	1,2	0,0	0,1	-1,7	11,2

Tabelle 4.11: Gewichtsmatrix C für 1-Chlor-2,4-dinitrobenzol mit zwölf Resonanzfrequenzen als Vorinformation.

Die Kreuzsignale im zugehörigen 2D-Spektrum (Abbildung (4.26a)) sind erwartungsgemäß von geringer Intensität. Aus dem 1D-Schnitt durch die Frequenzachse ω_1 bei Ω_5 in Abbildung (4.26b) ist ersichtlich, dass im Gegensatz zum 2D-Spektrum in Abbildung (4.25) die von parallelen Übergängen entstehenden Diagonalmultiplettsignale fehlen. Die Konstante J_{AX} kann weder über die im *E.COSY* mögliche aktive noch passive Koppelung bestimmt werden. Für die beiden anderen Konstanten sind $J_{AM} = 8,4$ Hz und $J_{AM} = 2,7$ Hz gemessen worden.

Durch Vorgabe aller Resonanzfrequenzen wird neben der auswertbaren Gewichtsmatrix C auch die Qualität des 2D-Spektrums erhöht. Andererseits sind durch Vorgaben von nur acht Resonanzfrequenzen die Konnektivitäten im 2D-Spektrum deutlicher zu erkennen.

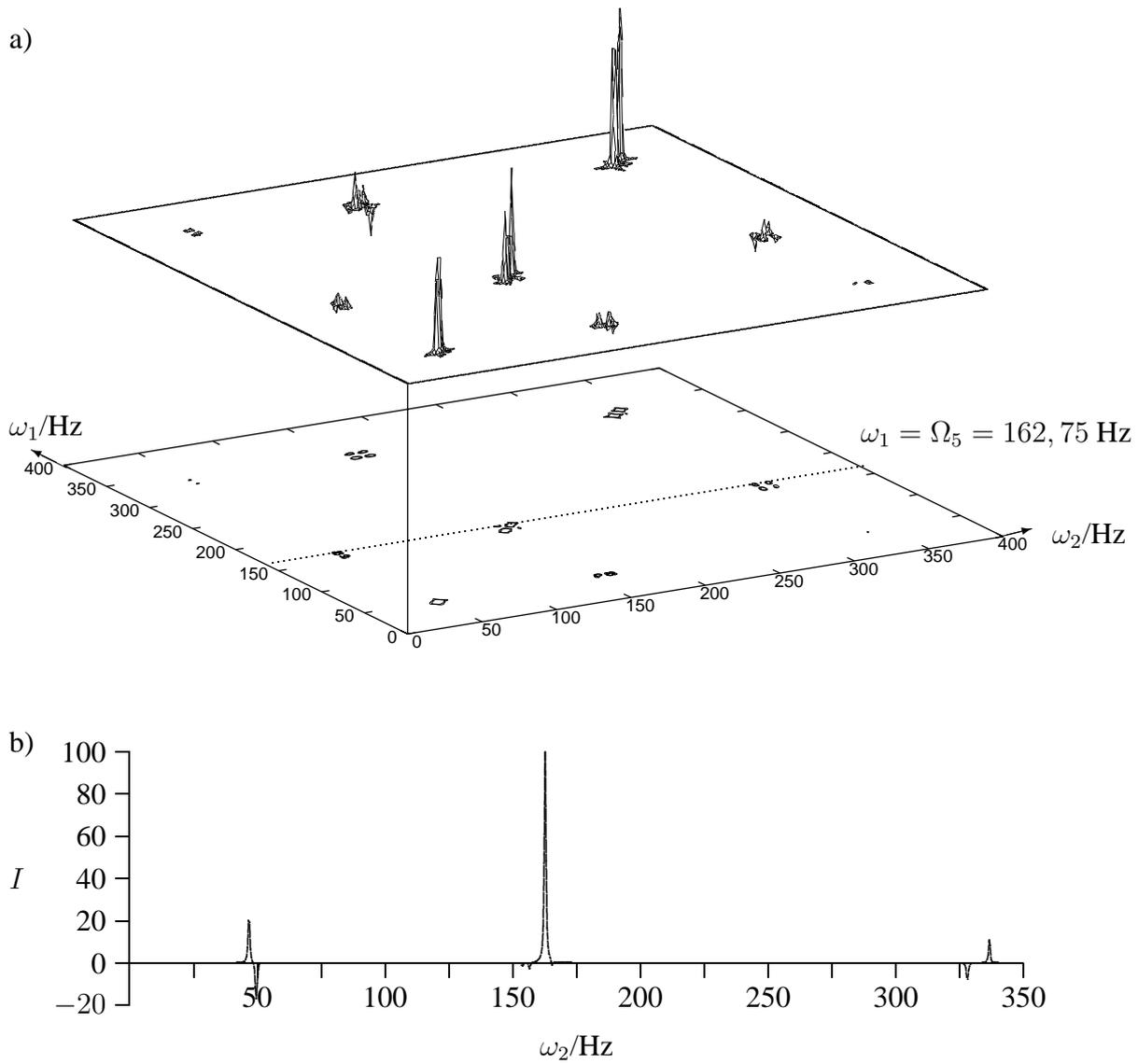


Abb. 4.26: a) Phasenkorrigierter 2D-Schnitt $K_3(\omega_1, \omega_2, -\omega_2)$ von 1-Chlor-2,4-dinitrobenzol mit zwölf Resonanzfrequenzen als Vorinformation und b) 1D-Schnitt durch das 2D-Spektrum bei $\omega_1 = \Omega_5 = 162,75 \text{ Hz}$.

5 Zusammenfassung

Das dynamische Verhalten $y(t)$ von Kernspinsystemen in Antwort auf die Einstrahlung einer Folge zufällig verteilter Hochfrequenzimpulse mittlerer Kippwinkel $x(t)$ wird mittels der nichtlinearen Systemanalyse gemäß $y(t) = \Phi[x(t)]$ behandelt. Einem 1993 erstmals vorgeschlagenen Ansatz folgend wird das Funktional Φ , eine unendliche Integralentwicklung mit den Volterra-Kernen $k_n(\tau_1, \dots, \tau_n)$, durch ein neuronales Netz (*NN*) approximiert. Diese *NN*-NMR lässt eine schnelle, messfehlertolerante ^1H -NMR-Strukturanalyse – mit minimaler Probenverweilzeit im Spektrometer und chemometrischem Höchstmaß (Nutzung von Rechnerclustern) – erwarten und bietet eine weitestgehend automatisierbare Alternative zur konventionellen zweidimensionalen (2D) NMR-Spektroskopie. Die Untersuchungen dieser Arbeit richteten sich folglich auf die Verbesserung der Netztopologien und Trainingsmethoden sowie die Entwicklung neuartiger Ansätze, die u.a. auch Vorinformationen einzubinden gestatten.

Die quantenmechanische Theorie der Kernspins ist so umfassend entwickelt, dass nahezu jedes Experiment rein rechnerisch abgebildet werden kann. Die Untersuchungen wurden deshalb schwerpunktmäßig per Simulation durchgeführt, da so apparaturbedingte Artefakte bei der Lösung prinzipieller Fragestellungen keinen Einfluss nehmen konnten. Experimente dienten lediglich zur Verifikation der Simulationsergebnisse.

Es zeigt sich, dass eine dynamische Anpassung der Netztopologie durch Herausnahme von Neuronenverbindungen kleinen Gewichts während des *NN*-Trainings möglich ist, was bei Spinsystemen 1. Ordnung zu kürzeren Rechenzeiten ohne gravierenden Qualitätsverlust der Spektren führt. Weiter ist es Erfolg versprechend, beim *NN*-Training zusätzlich die Zeitableitungen von Anregung und Signal zu verwenden. Dies muss - startend und endend mit x, y - alternierend mit \dot{x} und \dot{y} geschehen. Die Überwindung lokaler Minima verbessert sich damit signifikant.

Die Elemente der Gewichtsmatrix (nach dem Training) bei fester Netztopologie repräsentieren das jeweilige Spinsystem. Startend von einer Zufallsverteilung, entstehen sie iterativ durch Anpassung des Netzausgangs an $y(t)$. Hierfür existiert kein Algorithmus, welcher Robustheit mit hoher Geschwindigkeit vereinigt. Beim *CC*-Algorithmus steht einer sich automatisch angepassten Netztopologie ein umfangreicher Satz von Parametern gegenüber, deren Wahl die erreichbare Konvergenz stark beeinflusst. Hier war es bei größeren Spinsystemen häufig nicht möglich, das Training in akzeptabler Zeit erfolgreich zu beenden. Der *RPROP*-Algorithmus bietet eine robuste, fast parameterlose Trainingsalternative, erwies sich allerdings ebenfalls als zu rechenintensiv. Nach wie vor zeigte es sich, dass allein der *backpropagation*-Algorithmus die gewünschte Generalisierungsfähigkeit, Robustheit und Zeiteffizienz zu einem akzeptablen Kompromiss vereint.

Die Vorgabe der Resonanzlinien eines 1D-Spektrums stellte sich als erfolgreichste Variante der *NN*-NMR heraus. Selbst bei Spektren über einen großen Spektralbereich bleibt die Zahl der Neuronen der Zwischenschicht innerhalb jenes Bereichs, der ein erfolgreiches *NN*-Training zeitlich zulässt. Somit sind große Spinsysteme der *NN*-Signalanalyse zugänglich. Die Überlagerung mehrerer kleinerer Spinsysteme vermag das Netz ebenso zu erkennen. Selbst überlappte Resonanzsignale, erkennbar an Linienverbreiterungen, werden angezeigt, wenn nur die Zahl der Neuronen entsprechend erhöht wird und die Resonanzfrequenzen nahezu gleichbleibe. Hierbei bietet das Verfahren hinreichende Robustheit, um irrtümlich angenommene Überlagerungen auszuschließen. Umgekehrt resultieren auch Strukturinformationen, selbst wenn nicht alle Resonanzlinien eines Spinsystems diesem Netz vorgehalten werden. So konnte das stark koppelnde AA'BB'C-Spinsystem in Phenol mit relativ wenigen der eigentlich vorhandenen Resonanzfrequenzen derart analysiert werden, dass die para-, meta- und ortho-Protonen dem Phenol zugeordnet werden konnten. Eine Erweiterung des Verfahrens auf komplexe Gewichte gestattet die Generierung von 2D-Spektren in t_1 -Quadratur.

Die Vorgabe der Resonanzlinien eines 1D-Spektrums hat einen weiteren großen Vorteil. Die *NN*-Gewichtsmatrix c_{il} zeigt durch Absolutwert und Vorzeichen ihrer Nichtdiagonalelemente direkt die Konnektivitäten des Spinsystems an, wenn zumindest die intensitätsstärksten Einquantenübergänge dem Netz als Vorinformation angegeben wurden. Liegt darüber hinaus ein vollständiger Satz der Einquantenübergänge vor, so wird von der Gewichtsmatrix sogar die Multiplettstruktur der Kreuzsignale des 2D-Spektrums korrekt angezeigt. Selbst kleine Fernkopplungen, welche wegen der ubiquitären Linienverbreiterung zu keiner beobachtbaren Aufspaltung führen, sind dennoch über ihre Gewichtswerte erkennbar. Insgesamt ist festzuhalten, dass die übliche Auswertung eines 2D-NMR-Spektrums per Inspektion oder Mustererkennungsprogramm mit der *NN*-NMR entbehrlich und somit der Strukturzuordnungsprozess in seiner Gesamtheit im Rechner abbildbar wird (vollautomatische ^1H -NMR-Strukturaufklärung).

Ausgehend von der bekannten Bruttoformel einer reinen Substanz sind heute mit einem Strukturgenerator alle möglichen Strukturisomeren im Rechner vorhaltbar. Ihre Zahl ist zwar für übliche Molekülgrößen sehr groß, kann jedoch gemäß der Vorgeschichte und der bereits bekannten Eigenschaften der Substanz (z.B. der Verteilungskoeffizienten) wesentlich eingeschränkt werden. Es zeigt sich, dass die *NN*-NMR in der Lage ist, die Restmenge weiter zu reduzieren oder sogar eine eindeutige Strukturaussage zu liefern. Dazu werden die *NN*-Antworten aller in Frage kommenden Strukturisomere mit Hilfe der heute recht genau bekannten oder quantenchemisch berechenbaren Spinparameter simuliert. Ihre Korrelation mit der *NN*-Antwort der Messproben ist bereits während der Messung durchführbar. Nichtzutreffende Strukturen sind so frühzeitig ausschließbar. Unbedingte Voraussetzung dafür ist aber, wie die Untersuchungen zeigen, dass der lineare Anteil der simulierten und gemessenen *NN*-Signale vor der Berechnung des Korrelationskoeffizienten rechnerisch entfernt ist. Diese Maßnahme stellt sicher, dass Ungenauigkeiten der verwendeten Spinparameter (Lösungsmittel- und Temperatureinflüsse, Schwankungsbreite der Tabellenwerte oder QM-Rechnungen) weitestgehend an Gewicht verlieren. Diese Variante der *NN*-NMR hat ein hohes Potenzial in der kombinatorischen Chemie, bei der die Strukturaufklärung nach dem Ausschlussprinzip und nicht durch Bestätigung einer vorgegebenen Zielstruktur vorzunehmen ist.

Literaturverzeichnis

- [1] M. Schmidt und D. Ziessow, *Ber. Bunsenges.* **91**, 1110 (1987).
- [2] M. Schmidt, *Dissertation*, Technische Universität Berlin, D83 (1989).
- [3] P. Forster und D. Ziessow, *Proceedings of the 7th Workshop „Computer in Chemistry“*, D. Ziessow (Ed.), Gesellschaft Deutscher Chemiker Fachgruppe Chemie-Information-Computer, 119 (1993).
- [4] P. Forster, *Dissertation*, Technische Universität Berlin, D83 (1994).
- [5] P. Forster und D. Ziessow, *Proceedings of the 10th Workshop „Computer in Chemistry“*, J. Gasteiger (Ed.), Gesellschaft Deutscher Chemiker Fachgruppe Chemie-Information-Computer, 323 (1996).
- [6] B. Blümich and D. Ziessow, *J. Chem. Phys.* **78**, 1059 (1983).
- [7] B. Blümich and D. Ziessow, *Molec. Phys.* **48**, 955 (1983).
- [8] T. Kaulisch, H. Kolbe, and D. Ziessow *J. Magn. Reson. A* **121**, 42 (1996).
- [9] P. Arlt und D. Ziessow, *Proceedings of the 7th Workshop „Computer in Chemistry“*, D. Ziessow (Ed.), Gesellschaft Deutscher Chemiker Fachgruppe Chemie-Information-Computer, 15 (1993).
- [10] P. Arlt, *Dissertation*, Technische Universität Berlin, D83 (1994).
- [11] C. Tegeler, *Dissertation*, Technische Universität Berlin, D83 (1996).
- [12] M. Otto, *Chemometrie VCH*, Weinheim (1997).
- [13] R. R. Ernst, G. Bodenhausen and A. Wokaun, *Principles of Nuclear Magnetic Resonance in One and Two Dimensions*, Clarendon Press, Oxford (1987).
- [14] C. Griesinger, O. W. Sørensen, and R. R. Ernst, *J. Magn. Reson.* **84**, 14 (1989).
- [15] B. Blümich und D. Ziessow, *Ber. Bunsenges.* **84**, 1090 (1980).
- [16] B. Blümich, *Molec. Phys.* **48**, 969 (1983).
- [17] W. P. Aue, E. Bartholdi, and R. R. Ernst, *J. Chem. Phys.* **64**, 2229 (1976).
- [18] H. Kessler, M. Gehrke und C. Griesinger, *Angew. Chemie* **100**, 507 (1988).

- [19] M. Eberstadt, G. Gemmecker, D. F. Mierke und H. Kessler *Angew. Chemie* **107**, 1813 (1995).
- [20] J. Jeener, *Ampère International Summer School*, Basko Polje (1971).
- [21] O. W. Sørensen, G. W. Eich, M. H. Levitt, G. Bodenhausen, and R. R. Ernst, *Prog. Nucl. Magn. Reson. Spectrosc.* **16**, 163 (1983).
- [22] C. Bauer, R. Freeman, T. Frenkiel, J. Keeler, and A. J. Shaka, *J. Magn. Reson.* **58**, 442 (1984).
- [23] J. Keeler and D. Neuhaus, *J. Magn. Reson.* **63**, 454 (1985).
- [24] D. Marion and K. Wüthrich, *Biochem. Biophys. Res. Commun.* **113**, 967 (1983).
- [25] D. J. States, R. A. Haberkorn, and D. J. Ruben *J. Magn. Reson.* **48**, 256 (1982).
- [26] V. Volterra, *Theory of Functionals*, Dover, New York (1959).
- [27] R. R. Ernst, *J. Magn. Reson.* **3**, 10 (1970).
- [28] R. Kaiser, *J. Magn. Reson.* **3**, 28 (1970).
- [29] C. Griesinger, O. W. Sørensen, and R. R. Ernst, *J. Am. Chem. Soc.* **107**, 6394 (1985).
- [30] H. Armitage, U. Blömer, I. Genge, A. Khuen, E. E. Wille und D. Ziessow, *Ber. Bunsenges.* **91**, 1115 (1987).
- [31] B. Blümich, *Dissertation*, Technische Universität Berlin, D83 (1980).
- [32] B. Blümich and D. Ziessow, *J. Magn. Reson.* **46**, 385 (1982).
- [33] W. S. McCulloch and W. Pitts, *Bulletin of Math. Biophys.* **5**, 115 (1943).
- [34] J. Hertz, A. Krogh, and R. G. Palmer, *Introducing to the Theory of Neuronal Computing*, Addison-Wesley Publishing Company, Redwood City (1991).
- [35] X. Jain, X. Mao, and X. Mohiuddin, *Neurocomputing* **6**, 1 (1996).
- [36] J. Gasteiger und J. Zupan, *Angew. Chemie* **105**, 510 (1993).
- [37] J. Zupan und J. Gasteiger, *Neural Network for Chemists: A Textbook*, VCH-Publisher, Weinheim (1993).
- [38] P. Werbos, *Ph.D. thesis*, Harvard University, Cambridge, MA (1974).
- [39] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, *Nature* **333**, 533 (1986).
- [40] D. Soloway and J. Bialasiewicz, *Proceedings of the 1992 IEEE International Symposium on Intelligent Control*, 11 (1992).
- [41] A. Schäfer, *Dissertation*, Technische Universität Berlin, D83 (1989).
- [42] S. A. Smith, T. O. Levante, B. H. Meier, and R. R. Ernst, *J. Magn. Reson. A* **106**, 75 (1994).

-
- [43] A. Zell, T. Korb, T. Sommer, and R. Bayer, *Netsim: Ein Simulator für neuronale Netze*, GWAI-89, Springer-Verlag (1989).
- [44] A. Zell, G. Mamier, M. Vogt, and N. Mache, *Der Stuttgarter Neuronale Netze Simulator*, G. Dorffner, K. Möller, G. Paaß, S. Vogel (Hrsg.): *Konnektionismus und Neuronale Netze*, GMD-Studien **272**, 335 (1995).
- [45] Brügel, *Handbook of NMR Spectral Parameters*, **Vol. 1**, Tables 1-36, Heyden (1979).
- [46] R. L. Vold, J. S. Waugh, M. P. Klein, and D. E. Phelps, *J. Chem. Phys.* **48**, 3831 (1968).
- [47] A. E. Derome, *Modern NMR Techniques for Chemistry Research*, Pergamon, Oxford (1987).
- [48] T. Poggio and F. Girosi, *A Theory of Networks for Approximation and Learning*, A. I. Memo, **1140**, MIT (1989).
- [49] M. Riedmiller and H. Braun, *A Direct Adaptive Method for Faster Backpropagation Learning: The RPROP Algorithm*, H. Ruspini (Ed.): *Proceedings of the IEEE International Conference on Neural Networks*, 586 (1993).
- [50] S. E. Fahlmann and C. Lebiere, *The Cascade-Correlation Architecture*, D. S. Touretzky: *Advances in Neural Information Processing Systems 2*, 524 (1990).
- [51] S. E. Fahlmann and S. Baluja, *Reducing Network Depth in the Cascade-Correlation Learning Architecture*, Technical Report CMU-CS-94-209, Carnegie Mellon University (1994).
- [52] C. Wehrfritz, *Diplomarbeit*, Friedrich-Alexander-Universität Erlangen, Lehrstuhl für Statistik 1 (1994).
- [53] <http://gamma.magnet.fsu.edu>
- [54] <ftp://gamma.magnet.fsu.edu/pub/gamma>
- [55] <http://www.acdlabs.com>
- [56] <http://www-ra.informatik.uni-tuebingen.de/SNNS>.
- [57] <ftp://ftp.informatik.uni-tuebingen.de/pub/SNNS>
- [58] <http://www.gnu.org/software/gcc/gcc.html>
- [59] <http://www.gnu.org/software/binutils/binutils.html>

Anhänge

A.1 Verwendete NMR-Parameter

A.1.1 Verwendetes Spinsystem zur Überprüfung des *pruning*-Verfahrens

$$SW = 100 \text{ Hz} \quad \Delta t = 5 \text{ ms} \quad T_1 = T_2 = 0,4 \text{ s}$$

AX + P – Spinsystem	ABC + P – Spinsystem
$\nu_A = 15 \text{ Hz}$	$\nu_A = 10 \text{ Hz} \quad J_{AB} = 3 \text{ Hz}$
$\nu_X = 85 \text{ Hz}$	$\nu_B = 33 \text{ Hz} \quad J_{AC} = 6 \text{ Hz}$
$\nu_P = 35 \text{ Hz}$	$\nu_C = 50 \text{ Hz} \quad J_{BC} = 10 \text{ Hz}$
$J_{AX} = 6 \text{ Hz}$	$\nu_P = 80 \text{ Hz}$

A.1.2 Verwendete Spinsysteme zur Überprüfung des *RPROP*-Algorithmus

$$SW = \pm 50 \text{ Hz} \quad \Delta t = 10 \text{ ms} \quad T_1 = T_2 = 0,4 \text{ s}$$

AX + P – Spinsystem
 $\nu_A = -35 \text{ Hz} \quad \nu_X = 10 \text{ Hz} \quad \nu_P = 30 \text{ Hz} \quad J_{AX} = 5 \text{ Hz}$

A.1.3 Konnektivitätsbestimmung zur Trennung von überlagerten Spinsystemen ohne Überlappung

$$SW = 500 \text{ Hz} \quad \Delta t = 1 \text{ ms} \quad T_1 = T_2 = 0,4 \text{ s}$$

AX – Spinsystem	A'X' – Spinsystem	AMX – Spinsystem
$\nu_A = 30 \text{ Hz}$	$\nu_{A'} = 133 \text{ Hz}$	$\nu_A = 65 \text{ Hz} \quad J_{AM} = 15 \text{ Hz}$
$\nu_X = 400 \text{ Hz}$	$\nu_{X'} = 430 \text{ Hz}$	$\nu_M = 180 \text{ Hz} \quad J_{AX} = 5 \text{ Hz}$
$J_{AX} = 6 \text{ Hz}$	$J_{A'X'} = 10 \text{ Hz}$	$\nu_X = 350 \text{ Hz} \quad J_{MX} = 10 \text{ Hz}$

A.1.4 Konnektivitätsbestimmung zur Trennung von überlagerten Spinsystemen mit überlappenden Signalen

$$SW = 100 \text{ Hz} \quad \Delta t = 5 \text{ ms} \quad T_1 = T_2 = 0,4 \text{ s}$$

AX – Spinsystem	A'X' – Spinsystem	P – Spinsystem
$\nu_A = 15 \text{ Hz}$	$\nu_{A'} = 35 \text{ Hz}$	$\nu_P = 65 \text{ Hz}$
$\nu_X = 90 \text{ Hz}$	$\nu_{X'} = 83 \text{ Hz}$	
$J_{AX} = 7 \text{ Hz}$	$J_{A'X'} = 7 \text{ Hz}$	

A.1.5 Konnektivitätsbestimmung von 4-Aminobuttersäure

Die folgenden chemischen Verschiebungen und die Koppelungskonstanten von diesem und den darauf folgenden Spinsystemen sind mit dem Programm *ACD/HNMR* abgeschätzt und die Resonanzen für eine Spektrometerfrequenz von 400 MHz berechnet worden.

$$SW = \pm 250 \text{ Hz} \quad \nu_{\text{Offset}} = 840 \text{ Hz} \quad \Delta t = 2 \text{ ms} \quad T_1 = T_2 = 0,4 \text{ s}$$

A ₂ M ₂ X ₂ + P – Spinsystem			
$\nu_A = -192,0 \text{ Hz}$	$\nu_M = 80,0 \text{ Hz}$	$\nu_X = 224,0 \text{ Hz}$	$\nu_P = -76,0 \text{ Hz}$
$J_{AM} = 7,07 \text{ Hz}$	$J_{AX} = 6,88 \text{ Hz}$		

A.1.6 Konnektivitätsbestimmung von Phenol mit einem Kontrollsignal

$$SW = 250 \text{ Hz} \quad \nu_{\text{Offset}} = 2700 \text{ Hz} \quad \Delta t = 2 \text{ ms} \quad T_1 = T_2 = 0,4 \text{ s}$$

AA'BB'X – Spinsystem			P – Spinsystem
$\nu_A = 30,0 \text{ Hz}$	$\nu_X = 70,0 \text{ Hz}$	$\nu_B = 193,0 \text{ Hz}$	$\nu_P = 100,0 \text{ Hz}$
$J_{AA'} = 2,71 \text{ Hz}$	$J_{AX} = 1,09 \text{ Hz}$	$J_{AB} = 8,17 \text{ Hz}$	
$J_{AB'} = 0,49 \text{ Hz}$	$J_{A'B} = 0,49 \text{ Hz}$	$J_{A'B'} = 8,17 \text{ Hz}$	
$J_{BB'} = 1,74 \text{ Hz}$	$J_{BX} = 7,42 \text{ Hz}$		

A.1.7 Spinsysteme mit der Summenformel C₄H₆

$$SW = 2500 \text{ Hz} \quad \nu_{\text{Offset}} = 200 \text{ Hz} \quad \Delta t = 0,2 \text{ ms} \quad T_1 = T_2 = 0,8 \text{ s}$$

- Bicyclo[1.1.0]butan

AA'X ₄ – Spinsystem	
$\nu_A = 915,08 \text{ Hz}$	$\nu_X = 715,8 \text{ Hz}$
$J_{AA'} = 7,05 \text{ Hz}$	$J_{AX} = J_{A'X} = 1,5 \text{ Hz}$

- But-1-in

A ₃ M ₂ X – Spinsystem		
$\nu_A = 228,0 \text{ Hz}$	$\nu_M = 660,0 \text{ Hz}$	$\nu_X = 564,0 \text{ Hz}$
$J_{AM} = 7,4 \text{ Hz}$	$J_{AX} = 0,25 \text{ Hz}$	$J_{MX} = 2,6 \text{ Hz}$

- But-2-in

A_6 – Spinsystem

$$\nu_A = 523,47 \text{ Hz}$$

- Buta-1,2-dien

A_3MX_2 – Spinsystem

$$\nu_A = 436,0 \text{ Hz} \quad \nu_M = 1776 \text{ Hz} \quad \nu_X = 1600 \text{ Hz}$$

$$J_{AM} = 7,1 \text{ Hz} \quad J_{AX} = 3,45 \text{ Hz} \quad J_{MX} = -6,67 \text{ Hz}$$

- Buta-1,3-dien

$AA'BB'XX'$ – Spinsystem

$$\nu_A = 1824,0 \text{ Hz} \quad \nu_B = 1864,0 \text{ Hz} \quad \nu_X = 2308,0 \text{ Hz}$$

$$J_{AA'} = 1,3 \text{ Hz} \quad J_{AB} = 1,85 \text{ Hz} \quad J_{AB'} = 0,86 \text{ Hz} \quad J_{AX} = 10,17 \text{ Hz}$$

$$J_{AX'} = -0,86 \text{ Hz} \quad J_{A'B} = 0,6 \text{ Hz} \quad J_{A'B'} = 1,91 \text{ Hz} \quad J_{A'X} = -0,86 \text{ Hz}$$

$$J_{A'X'} = 10,17 \text{ Hz} \quad J_{BB'} = 0,69 \text{ Hz} \quad J_{BX} = 17,05 \text{ Hz} \quad J_{BX'} = -0,83 \text{ Hz}$$

$$J_{B'X} = -0,83 \text{ Hz} \quad J_{B'X'} = 17,05 \text{ Hz} \quad J_{XX'} = 10,41 \text{ Hz}$$

- Cyclobuten

$A_2A'_2XX'$ – Spinsystem

$$\nu_A = 1027,92 \text{ Hz} \quad \nu_X = 2480,4 \text{ Hz}$$

$$J_{AA'} = -12,0 \text{ Hz} \quad J_{AX} = -0,8 \text{ Hz} \quad J_{AX'} = 1,55 \text{ Hz} \quad J_{A'X} = -0,8 \text{ Hz}$$

$$J_{A'X'} = 1,55 \text{ Hz} \quad J_{XX'} = 2,7 \text{ Hz}$$

- Metylcyclopropan

A_4XX' – Spinsystem

$$\nu_A = 183,91 \text{ Hz} \quad \nu_M = 2107,76 \text{ Hz}$$

$$J_{AX} = 2,63 \text{ Hz} \quad J_{AX'} = 2,63 \text{ Hz} \quad J_{XX'} = 1,15 \text{ Hz}$$

A.2 Optimierung der Programme

A.2.1 Optimierung durch Compilerschalter

Die größte Zeitminimierung bringen Optimierungsschalter im Compiler selbst. Allein das Kompilieren mit diesen Schaltern verkürzt die Rechenzeit um über ein Drittel. Es wurde auf einem PC mit Intel-Architektur mit dem C-Compiler *gcc* (Version 2.95.2 [58]) folgende Optimierungen angewandt:

```
gcc -O2 -fomit-frame-pointer -pipe -fno-strength-reduce \
-malign-loops=2 -malign-jumps=2 -malign-functions=2
```

Der spezielle Optimierungsschalter `-O2`, steht als Synonym im *gcc* für die folgenden Schalter:

```
-fdefer-pop -fcse-follow-jumps -fcse-skip-blocks \
-fthread-jumps -fexpensive-optimizations -fpeephole \
-fforce-mem -ffunction-cse -finline -fcaller-saves \
-fpcc-struct-return -frerun-cse-after-loop \
-fcommon -fgnu-linker -m80387 -mhard-float -mno-soft-float \
-mno-386 -m486 -mieee-fp -mfp-ret-in-387A
```

Diese Schalter nehmen dem Programmierer aber nicht jegliche Verantwortung für überflüssige Schleifen oder unnütze Parameterübergaben ab. Eine genauere Analyse zur Geschwindigkeitsoptimierung fand nach der Kompilation mit dem Schalter `-pg` anhand des Programms *gprof* (Version 2.9.5 [59]) statt. Dieses erlaubt die sukzessive Verfolgung jedes einzelnen Funktionsaufrufs. Es werden graphisch die Anzahl der Funktionsaufrufe dargestellt, die Aufrufreihenfolge und die Hierarchie, sowie die Zeit, die für die Ausführung dieser Funktionen benötigt werden.

Anhand *gprof* konnten dann die zeitintensivsten Schritte bei der Ausführung für die Netze untersucht werden. Die drei zeitaufwändigsten Schritte sind dabei die Abfrage der Anregungswerte, die Berechnung der Systemantwort durch das neuronale Netz, sowie die Anpassung der Gewichte mit Hilfe des Lernalgorithmus. Vor allem der letzte Schritt nimmt 60% der gesamten Rechenzeit in Anspruch und wurde daher besonders optimiert.

A.2.2 Optimierung des Quellcodes

Mit gesetzten Optimierungsschaltern sind weitere Versuche unternommen worden, das Laufzeitverhalten im rechenintensiven Trainingsprogramm zu verbessern. Da die entstanden Zeitoptimierungen stark vom ausgehenden nicht optimierten Programmcode abhängen, wird auf Zeitvergleiche verzichtet.

- Ein großer Teil an Rechenzeit wird vermieden, wenn innerhalb von Funktionen die Parameteranzahl klein gehalten wird. Ein konsequentes Vermeiden jeglicher zusätzlicher Variablen lässt dem Compiler keine Freiheiten mehr an „unnützen“ Stellen Optimierungen vorzunehmen. Das mag trivial klingen, jedoch werden allzu häufig zur Übersichtlichkeit des Programmcodes Zwischenwerte ausgerechnet oder Hilfszähler angelegt. Der daraus entstandene Zeitgewinn rechtfertigt jedoch bei zeitkritischen Funktionen eine mangelnde Übersichtlichkeit.

- Alle Schleifen wurden in separate Funktionen geschrieben. Dabei wurden immer nur die Parameter übergeben, die tatsächlich gebraucht wurden. Bei kleineren Funktionen wurde dies als so genannte *inline*-Funktion eingebaut. Um die Größe des Programmcodes klein zu halten, wird weiterhin auf die Struktur der Schleifen geachtet. Weisen Schleifen gleiche Muster auf, so werden sie zusammengefasst und gesondert programmiert, auch wenn sie verschiedene physikalische Bedeutungen beinhalten.

Der Preis für diese Zeitoptimierung ist auch hier die Leserlichkeit des Programmcodes. Zum einen tauchen die Schleifen an anderen Stellen bzw. Dateien auf, zum anderen wirkt die Übergabe von beispielsweise sieben einzelnen Parametern anstelle eines strukturierten Gesamtparameters verwirrend.

- Der Anteil der Berechnung der Transferfunktion und seiner Ableitung, die im Laufe des Trainings kontinuierlich benötigt wird, betrug anteilig zur Gesamtrechenzeit zwar nur 3%, jedoch wurde auch hier eine Optimierung getestet:

Anstelle das Programm immer wieder Funktionsabbildungen berechnen zu lassen, wurden alle möglichen Abbildungen für die Transferfunktion und dessen Ableitung für ein geeignetes Intervall in den Arbeitsspeicher abgelegt. Da das Programm mit einer Genauigkeit der *floating point precision* arbeitet, stand zur Wahl, sämtliche Werte abzulegen, die dieser Genauigkeit entsprechen, oder eine Teilmenge davon. Letzteres hätte eine lineare Interpolation zur Folge, falls Werte außerhalb der Teilmenge im Training benötigt würden. Beide Möglichkeiten unterscheiden sich in der Zeit der Wertinitialisierung, sowie an dem Platz, der im Arbeitsspeicher dazu benötigt wird. Der Geschwindigkeitsvorteil für eine komplette Vorausberechnung überwog, sodass diese Möglichkeit als Erste gewählt wurde.

Untersucht wurde der Sachverhalt anhand der rechenintensiven Transferfunktionen $\mathfrak{f}(z) = ze^{-z^2}$ und seiner Ableitung $\mathfrak{f}'(z) = (1 - 2z)e^{-z^2}$. Die Analyse zeigt, dass das Berechnen und Schreiben der Funktionsabbildungen in den Adressierungsbereich des Arbeitsspeichers kaum Zeit in Anspruch nimmt. Die Adressberechnung für den Zugriff auf den Wert im Speicher nimmt jedoch mehr Zeit in Anspruch als eine direkte Berechnung; das Programm lief langsamer.

Die zweite Möglichkeit, die durch die Interpolation noch zusätzliche Rundungsfehler produziert, lief erwartungsgemäß ebenfalls langsamer als die direkte Funktionsberechnung.

- In den komplexen Netzen werden bei den diversen mathematischen Operationen die Real- und Imaginärteile nacheinander bearbeitet. So werden z.B. bei einer Summation von zwei komplexen Größen in C erst die beiden Realteile und anschließend die beiden Imaginärteile summiert. Dies erfolgt in getrennten Rechenschritten.

An dieser Stelle bietet sich der Einsatz von $C++$ an, da es in dieser Programmiersprache möglich ist, weitere Operatorenvorschriften zu den Standardoperatoren zu definieren. Dieses so genannte Überladen von Operatoren ermöglicht die Definition von arithmetischen Operationen für komplexe Größen.

Die Untersuchung zeigte, dass mit dem Einsatz von $C++$ kein Vorteil zur Geschwindigkeitsoptimierung erzielt werden konnte. Lediglich die Übersichtlichkeit des Programmcodes steigerte sich, da nun die einzelnen arithmetischen Operationen für Real- und Imaginärteil nicht mehr aufgeteilt werden mussten.

Anhänge

Der absolute Zeitgewinn ist selbst bei geringfügiger Optimierung beträchtlich. Ein neuronales Netz mit 40 Neuronen in der Zwischenschicht mit der Transferfunktion $\mathfrak{f}(z) = ze^{-z^2}$ und 2048 Neuronen in der Eingabeschicht wurde mit 500.000 Werten der Systemantwort des 1-Chlor-2,4-dinitrobenzols mit der bestmöglichen Optimierung trainiert. Die Rechenzeit auf einem Rechner mit zwei *Intel 500 MHz Pentium III*-Prozessoren betrug neun Stunden.

Die Ausnahmen langer Rechenzeiten bilden die Netze, die Vorinformationen bekommen. Diese Netze erlernen den Zusammenhang zwischen Spinsystemanregung und -antwort teilweise in wenigen Sekunden. Dennoch wurden auch für diese Netze die herausgefundenen Optimierungsmöglichkeiten verwendet.

A.2.3 Verteilung der Abläufe auf mehrere Rechnersysteme

Komplizierte Spinsysteme können durch digitale Filterung (Abschnitt 2.5.1) in Teilsysteme zerlegt und von unabhängigen neuronalen Netzen getrennt berechnet werden. Getrennte Rechnungen können auch durchgeführt werden, wenn verschiedene Lernparameter, Anfangsbedingungen oder Vorinformationen eingestellt werden.

Hier bietet sich der Einsatz von vernetzten Rechnern an. Die Anregungssequenz und die Antwort des Spinsystems liegen auf einem zentralen Rechner (*Server*) und können von verschiedenen Rechnern im Netzwerk (*Clients*) bearbeitet werden, ohne dass ein großer Datentransfer stattfinden muss. Jeder einzelne Rechner bearbeitet seinen Parametersatz und speichert sein Teilergebnis auf dem *Server* ab, auf dem dann die Auswertung stattfinden kann. Somit können ohne großen Aufwand Rechnungen parallelisiert werden, um die Analyse zu beschleunigen.

Bei der Verwendung von verschiedenen Rechnersystemen ist auf die Rechnerkompatibilität zu achten. Die Werte für die Anregung und Antwort sind als binäre Zahlen abgelegt, da das ASCII-Format ungefähr doppelt so viel Speicherplatz einnimmt. Die binären Werte haben auf Rechnern mit der so genannten *i386*-Prozessorarchitektur (z.B. *Intel-PC*) ein anderes Format, als auf Rechnern mit *RISC*-Prozessoren (z.B. *IBM*, *SUN*). Hier muss zunächst auf beiden Rechnern eine Konvertierung erfolgen.

A.3 RPROP-Algorithmus für die NMR-Signalanalyse

Die Fehlerfunktion E für ein Spinsystem mit der Gedächtnislänge L lautet für M Muster zum Training eines neuronalen Netzes mit I Zwischenschichtneuronen

$$E = \sum_{m=1}^M (y_m - \tilde{y}_m)^2 + \mu \cdot \left(\sum_{i=1}^I \sum_{l=1}^L c_{il}^2 + \sum_{i=1}^I v_i^2 \right) \quad (\text{A.1})$$

$$= \sum_{m=1}^M \left\{ y_m - \sum_{i=1}^I v_i \mathfrak{F}(z_i) \right\}^2 + \mu \cdot \left(\sum_{i=1}^I \sum_{l=1}^L c_{il}^2 + \sum_{i=1}^I v_i^2 \right) \quad (\text{A.2})$$

$$= \sum_{m=1}^M \left\{ y_m - \sum_{i=1}^I v_i \sum_{l=1}^L c_{il} x_{m-l+1} \right\}^2 + \mu \cdot \left(\sum_{i=1}^I \sum_{l=1}^L c_{il}^2 + \sum_{i=1}^I v_i^2 \right). \quad (\text{A.3})$$

Die Ableitungen ergeben sich zu

$$\frac{\partial E}{\partial c_{il}} = -2\mathfrak{F}'(z_i) v_i \sum_{m=1}^M (y_m - \tilde{y}_m) x_{m-l+1} + 2\mu c_{il} \quad (\text{A.4})$$

$$\frac{\partial E}{\partial v_i} = -2\mathfrak{F}(z_i) \sum_{m=1}^M (y_m - \tilde{y}_m) + 2\mu v_i. \quad (\text{A.5})$$

Da im *RPROP*-Algorithmus nur die Vorzeichen der Ableitungen von Interesse sind, kann der Faktor 2 in den Gleichungen (A.4) und (A.5) vernachlässigt werden. Die Berechnung der *update-values* und die Anpassung der einzelnen Gewichte nach den Gleichungen (4.16)-(4.18) erfolgt erst nach der Präsentation aller Muster. Aufgrund dieses *batch*-Trainings müssen die Daten pro Präsentation temporär angelegt werden, was im Gegensatz zum *backpropagation*-Algorithmus eine größere Anforderung an den Arbeitsspeicher des Rechners zur Folge hat. Für komplexe Größen erweitern sich die Ausdrücke zu

$$E = \Re \left\{ \sum_{m=1}^M (y_m - \tilde{y}_m)^2 \right\} + \Im \left\{ \sum_{m=1}^M (y_m - \tilde{y}_m)^2 \right\} + \quad (\text{A.6})$$

$$\mu \cdot \left(\Re \left\{ \sum_{i=1}^I \sum_{l=1}^L c_{il}^2 + \sum_{i=1}^I v_i^2 \right\} + \Im \left\{ \sum_{i=1}^I \sum_{l=1}^L c_{il}^2 + \sum_{i=1}^I v_i^2 \right\} \right). \quad (\text{A.7})$$

Die komplexen Ableitungen können entsprechend gebildet werden [11].

Als Ausgangswert für die *update-values* wurde $\Delta_{il} = 0,1$ gewählt. Als Obergrenze für diesen Parameter galt $\Delta_{\max} = 50$. Die Lernfaktoren wurden mit $\eta^- = 0,5$ und $\eta^+ = 1,2$ festgesetzt.

Lebenslauf

Adresse Greulichstr. 24, 12277 Berlin
E-mail sielaff@chem.tu-berlin.de
geboren am 28. Mai 1970 in Madrid (Spanien)
Staatsangehörigkeit Deutsch
Familienstand Verheiratet mit Heike Sielaff, geborene Enke

Schulbildung

1976 - 1982 Alt-Lankwitzer Grundschule Berlin-Steglitz
1982 - 1989 Luise-Henriette-Oberschule Berlin-Tempelhof
Juli 1989 Abitur

Universität

1989 - 1994 Chemiestudium an der Technischen Universität Berlin
Juli 1994 Diplomarbeit am Iwan-N.-Stranski-Institut für Physikalische und Theoretische Chemie unter Anleitung von Herrn Prof. Dr. D. Ziesow
Seit 1994 Arbeit an der vorliegenden Dissertation in der Arbeitsgruppe von Herrn Prof. Dr. D. Ziessow
1995 - 2000 Wissenschaftlicher Mitarbeiter am Iwan-N.-Stranski-Institut
Seit Juli 1999 Mitarbeit am BMBF-Leitprojekt *Vernetztes Studium – Chemie, Teilprojekt Physikalische Chemie und Mathematik*

Die vorliegende Arbeit wurde am Iwan-N.-Stranski-Institut für Physikalische und Theoretische Chemie unter Anleitung von Herrn

Prof. Dr. D. Ziessow

angefertigt. Ich bin ihm sehr dankbar, dass er neben der Betreuung meiner Dissertation mir viele Inspirationen gegeben und Gestaltungsfreiheit gewährt hat.

Ferner möchte ich danken:

Herrn Prof. Dr. G. H. Findenegg für die freundliche Übernahme des Koreferats,

Herrn Dr. T. Kaulisch für die vielen nützlichen (virtuellen und zwischenmenschlichen) Diskussionen und Anregungen,

Herrn Dipl.-Ing. T. Wawrzinek für die Hilfestellungen bei der Optimierung der Programme,

Herrn Dr. M. Kowall für diverse Ratschläge und konstruktive Kritik,

Allen nicht näher genannten Mitarbeitern des AK Ziessow und des INSI für die sehr angenehme und freundliche Arbeitsatmosphäre,

Meinen Eltern und Schwiegereltern für die vielen nichtwissenschaftlichen Unterstützungen.

A minha esposa Heike um grande agradecimento especial por sua protecção mental e sua compreensão permanente.