

**Ein konnektionistisches Modell von Arbeitsgedächtnis-
Effekten beim Menschen**

**Diplomarbeit zum Erlangen des akademischen Grades „Diplom-
Informatiker“
im Fachbereich Informatik
der
Universität Bremen**

**Mathias Niemann
Carl-Schurz-Straße 2e
28209 Bremen
31.03.2008 Bremen**

**Matrikelnummer
1481355**

**1. Gutachter: Dr. Thomas Barkowsky
2. Gutachterin: Prof. Dr. Kerstin Schill**

1	Einleitung.....	1
2	Untersuchungen der Mechanismen des Gedächtnisses.....	2
2.1	Verschiedene empirische Untersuchungsansätze	2
2.2	Das „immediate serial recall“	4
2.2.1	Der „list-length“-Effekt.....	4
2.2.2	Der „primacy“-Effekt	5
2.2.3	Der „recency“-Effekt	5
2.2.4	„Errors of omission“	6
2.2.5	„Errors of commission“	6
2.3	Zusammenfassung.....	7
3	Das Arbeitsgedächtnis und verschiedene aktuelle Modelle	8
3.1	Das “Multiple-Component Model” von Alan D. Baddeley und Robert H. Logie	10
3.1.1	Der „phonological loop“	12
3.1.2	Das „visuospatial sketchpad“	14
3.1.3	Der „episodic buffer“	15
3.1.4	Die „central executive“	16
3.1.5	Zusammenspiel der Komponenten bei kognitiven Aufgaben.....	17
3.1.6	Langzeitgedächtnis und Arbeitsgedächtnis	19
3.1.7	Bewertung des „Multiple-Component-Model“	20
3.2	Das „Embedded-Process Model” von Nelson Cowan	23
3.2.1	Mechanismen des Arbeitsgedächtnisses.....	25
3.2.2	Steuerung des Arbeitsgedächtnisses	27
3.2.3	Beschränkungen des Arbeitsgedächtnisses	28
3.2.4	Die Rolle des Langzeitgedächtnisses.....	29
3.2.5	Bewertung des Modells	30
3.3	„Working Memory in Everyday Skilled Performance” von K. Anders Ericsson und Peter F. Delaney	33
3.3.1	„Short Term – Working Memory”	34
3.3.2	Hinweise für zusätzliche Kapazitäten	34
3.3.3	„Long Term – Working Memory”	36
3.3.4	Bewertung des Modells	40

3.4	„A Recurrent Neural Network Model“ von Matthew M. Botvinick und David C. Plaut.....	42
3.4.1	Aufbau des Modells	43
3.4.2	Funktionsweise des neuronalen Netzes	44
3.4.3	Ergebnisse der Simulationen.....	46
3.4.4	Bewertung des Modells	47
3.5	Wichtige Probleme und Widersprüche aktueller Ansichten.....	51
4	Das Kanal-Modell des Arbeitsgedächtnisses.....	53
4.1	Biologische Grundlagen des Modells.....	53
4.1.1	Struktur des Nervensystems.....	54
4.1.2	Signalübertragung durch Transmitter und Rezeptoren.....	55
4.1.3	Verschiedene Mechanismen zur Informationsspeicherung	57
4.1.4	Entwicklung unterschiedlicher Areale im Gehirn.....	59
4.1.5	Zusammenfassung	60
4.2	Struktur und Mechanismen des Modells auf Basis der biologischen Grundlagen	61
4.2.1	Struktur.....	61
4.2.2	Mechanismen	61
4.3	Umsetzung des Modells als neuronales Netz für die Aufgabe des „immediate serial recall“	65
4.3.1	Aufbau des neuronalen Netzes	65
4.3.2	Input des Netzes.....	66
4.3.3	Berechnung eines Folgeschrittes	66
4.3.4	Lernfunktionen.....	67
4.3.5	Output des Netzes.....	70
4.3.6	Zusammenspiel der Komponenten.....	70
5	Evaluation	72
5.1	Modellvorhersagen.....	72
5.1.1	Der „list length“-Effekt.....	72
5.1.2	Der „primacy“-Effekt	73
5.1.3	Der „recency“-Effekt	73
5.1.4	„Errors of omission“	74
5.1.5	„Errors of commission“	75
5.2	Die Simulationen	76

5.2.1	Simulation 1	76
5.2.2	Simulation 2	78
5.2.3	Simulation 3	79
5.2.4	Simulation 4	80
6	Zusammenfassung und Ausblick	81
7	Literaturverzeichnis	82
8	Anhang	87
8.1	Quellcode	87
8.1.1	Start.m	87
8.1.2	loop.m	94
8.1.3	auto.m	96
8.1.4	calc.m	98
8.1.5	dwnet.m	99
8.1.6	learn.m	100
8.1.7	recall.m	101
8.1.8	recognize.m	102
8.1.9	randvec.m	103

Abbildungsverzeichnis

Abbildung 1: Anteil korrekter Wiedergaben von Listen für unterschiedliche Listenlängen (Crannell & Parrish, 1957).....	5
Abbildung 2: Verteilung der unterschiedlichen Fehlerarten bei Patienten mit geschlossenen Kopfverletzungen (Schwarz, Reed, Montgomery, Palmer, & Mayer, 1991)	6
Abbildung 3: Modell von Baddeley und Hitch von 1974 (Baddeley & Logie, 1999) ..	10
Abbildung 4: Arbeitsgedächtnismodell mit „episodic buffer“(Baddeley, 2000)	12
Abbildung 5: Arbeitsgedächtnismodell nach Cowan (Cowan, 1999)	24
Abbildung 6: Struktur der Abrufhinweise im LT-WM für eine Ziffernfolge von 30 Ziffern (Ericsson & Delaney, 1999).....	39
Abbildung 7: Anatomie einer Synapse (Pinel & Pauli, 2007)	56
Abbildung 8: Lernregel für die Übertragungsgewichte. Aktivierte Neuronen sind rot dargestellt während inaktive Neuronen weiß sind.	68
Abbildung 9: Zusätzliche Unterscheidung für die kurzfristigen Übertragungsgewichte.	69
Abbildung 10: Anteile der korrekt wiederholten Listen unterschiedlicher Länge	77
Abbildung 11: Durchschnittliche Ähnlichkeit der einzelnen Aktivierungsmustern zu den Repräsentationen der Listenelementen	78
Abbildung 12: Durchschnittliche Ähnlichkeit ohne Kurzzeit-Gewichte	79
Abbildung 13: Fehlerverhältnis bei Kanalgrößen von 50 % (links) und 40 % (rechts) der Schicht.....	80
Abbildung 14: Fehlerverhältnis bei Kanalgrößen von 30 % (links) und 20 % (rechts) der Schicht.....	80

Tabellenverzeichnis

Tabelle 1: Anteile der korrekt wiederholten Listen unterschiedlicher Länge	77
--	----

1 Einleitung

Die Funktionsweise des menschlichen Gehirns ist auch heute noch zu großen Teilen unbekannt. Um trotzdem Begründungen für bestimmte Verhaltensformen finden zu können, wurden Modelle entwickelt, die die Prozesse des Gedächtnisses abbilden. Im Rahmen dieser Diplomarbeit beschäftige ich mich mit verschiedenen Modellen, die versuchen die Mechanismen des Arbeitsgedächtnisses zu erklären.

Um verschiedene Eigenschaften des Arbeitsgedächtnisses zu definieren wurde eine Vielzahl unterschiedlicher Untersuchungsmethoden ausgearbeitet. Hierzu werde ich im folgenden Kapitel einen kurzen Überblick verschaffen, bevor ich mich der Untersuchungsmethode des „immediate serial recall“ zuwende. Anhand der Ergebnisse werde ich die Notwendigkeit der Hinterfragung aktueller Modelle aufzeigen. Ich werde mich dabei auf die Methode konzentrieren, die die Leistungen von Testpersonen bei der Aufgabe des „immediate serial recall“ untersucht.

Anschließend werde ich vier aktuelle Modelle des Arbeitsgedächtnisses vorstellen und untersuchen, in wie weit diese Modelle Probleme haben, die Ergebnisse der Untersuchungsmethode erklären zu können.

Im Kapitel 4 werde ich dann ein eigenes Modell einführen, welches diese Probleme der anderen Modelle nicht hat. Dieses eigene Modell wurde als neuronales Netz implementiert. Im Kapitel 5 werde ich Simulationen der „immediate serial recall“-Aufgabe beschreiben, die mit dem implementierten neuronalen Netz durchgeführt wurden. Anhand der Ergebnisse der Simulationen zeige ich, dass das von mir vorgestellte Modell die Effekte, die beim „immediate serial recall“ auftreten besser und leichter erklärt, als die anderen dargestellten Modelle.

2 Untersuchungen der Mechanismen des Gedächtnisses

Seit den ersten Untersuchungen des amerikanischen Physikers Francis Eugene Nipher (1876) wurden eine Vielzahl unterschiedliche Methoden angewandt, um die Mechanismen des Arbeitsgedächtnisses genauer zu erforschen. Nipher dokumentierte damals die Fehlerzahlen, die Testpersonen beim Wiederholen von Listen machten. Dabei stellte er fest, dass die Fehlerzahlen bei den ersten und letzten Elementen der Liste kleiner waren als die bei den Elementen in der Mitte der Liste (Nipher, 1876). Kurze Zeit später dokumentierte Ebbinghaus (1885) den Fortschritt, den er dabei machte, sich Listen sinnloser Silben einzuprägen, um somit Schlussfolgerungen über die Eigenschaften des menschlichen Lernens zu gewinnen. Die Lernkurve und Vergessenskurve, die er dabei entwickelte, werden noch heute verwendet.

Heutzutage existieren Dokumentationen einer breiten Palette von unterschiedlichen Herangehensweisen zur Ermittlung von Parametern und Mechanismen des Arbeitsgedächtnisses.

In diesem Kapitel werde ich einen Überblick über die Vielfältigkeit der unterschiedlichen Ansätze und deren Ergebnisse geben. Anschließend werde ich die spezielle Untersuchungsmethode, die ich zur Darlegung der Funktionsweise und der Unterschiede von den Arbeitsgedächtnismodellen dieser Arbeit gewählt habe, genauer erläutern.

2.1 Verschiedene empirische Untersuchungsansätze

Wie im vorigen Kapitel erwähnt, unterscheiden sich die Methoden zur Analyse der Kapazitäten und Mechanismen des Gedächtnisses stark, je nach Forschungsschwerpunkt.

Die Versuchsobjekte bzw. –subjekte können sich dabei in Menschen, Primaten und andere Lebewesen oder sogar in vitro kultivierte Nervenzellen unterscheiden.

Bei den Versuchen mit Lebewesen kann wiederum in gesunde Probanden und Testsubjekte mit Gedächtnisschädigungen separiert werden. Solche Schädigungen können durch Unfälle oder Eingriffe sowohl physisch als auch psychisch verursacht worden sein. Es können angeborene Krankheiten sein oder altersbedingte Abbauererscheinungen.

Bei Untersuchungen von Entwicklungsphasen werden Testsubjekte verschiedener Altersstufen gewählt.

Wenn allgemeingültige Aussagen getroffen werden sollen werden häufig große Gruppen unter gleichen Bedingungen getestet, um eine breite Basis vergleichbarer Daten zu bekommen.

Wie im Kapitel zum Modell von Ericsson und Kintsch (1995) (vgl. Kapitel 3.3) erwähnt, werden gelegentlich aber auch einzelne Personen mit besonders herausragenden Fähigkeiten eingehend untersucht, um die Ursachen dafür aufzudecken, dass ihr Verhalten sich deutlich von den Leistungen Anderer abhebt.

Die Auswertungen der genannten Untersuchungen können zum einen auf Grund der Reaktion oder Aussagen der Testpersonen und zum anderen durch bildgebende Verfahren anhand von Gehirnaktivitäten erfolgen. Meist wird bei der Auswertung jedoch versucht den Einfluss von Aussagen der Testperson zu minimieren, da unterschiedliches Urteilsvermögen in der Praxis nicht zu verlässlichen Ergebnissen führt (Baddeley, 1999).

Um vergleichbar mit dem Modell von Baddeley und Logie (1999) (vgl. Kapitel 3.1) das Arbeitsgedächtnis in verschiedene unabhängige Bereiche unterteilen zu können, werden Subjekte mit mehreren Aufgaben betraut, die gegebenenfalls zu gegenseitiger Beeinträchtigung führen oder auch nicht.

Die Untersuchungsmethode, die ich als gemeinsame Plattform für den Vergleich und die Beschreibung der Modelle wähle, ist das „*immediate serial recall*“. Dieses Verfahren ist meines Erachtens besonders geeignet. So ist die Anwendung dieses Verfahrens schon seit dem Ende des 19. Jahrhunderts dokumentiert. Seitdem hat diese Methode eine besonders hohe Bedeutung für die Untersuchung des Arbeitsgedächtnisses erlangt (Cowan, Saults, Alliot, & Moreno, 2002) und sie wurde in vielen verschiedenen Abwandlungen durchgeführt. Somit existiert eine große Menge von Testdaten.

Ein weiterer Aspekt, der für die Auswahl des „*immediate serial recall*“ spricht ist die Tatsache, dass die Ergebnisse dieser Methode, trotz ihrer unkomplizierten Beschreibung und Durchführung, mehrere verschiedene Effekte aufzeigen. Diese Effekte zeigen Eigenschaften des Arbeitsgedächtnisses, die die Modelle erklären können müssen.

Trotz der umfangreichen Menge an Untersuchungsergebnissen sind die zugrunde liegenden Mechanismen beim „*immediate serial recall*“ jedoch bis heute unklar. Mit meiner Arbeit beabsichtige ich, einen Beitrag zur Aufklärung dieser Mechanismen zu leisten.

2.2 Das „immediate serial recall“

Eine Untersuchungsmethode, die mittlerweile den Stellenwert eines Benchmark-Tests (Botvinick & Plaut, 2006) für Arbeitsgedächtnis-Modelle darstellt ist der „immediate serial recall“-Test.

Bei diesem Test wird den Testpersonen eine Abfolge von Buchstaben, Zahlen, Wörtern, Bildern oder Ähnlichem präsentiert, welche unmittelbar wiederholt werden müssen. Dabei unterscheidet sich das „*serial recall*“ zum „*free recall*“ indem die präsentierte Abfolge in derselben Reihenfolge wiederholt werden muss und nicht in beliebiger Reihenfolge, wie es beim „*free recall*“ der Fall ist. Außerdem besagt schon die Bezeichnung „immediate serial recall“-Test, dass zwischen Präsentation und Wiederholung der Liste üblicherweise weder eine Pause noch eine Ablenkungsaufgabe stattfindet.

Die Ursprünge des „immediate serial recall“-Tests gehen bis auf Nipher (1876) und kurz darauf Ebbinghaus (1885) zurück, die beide etwa zur gleichen Zeit ihre jeweiligen Dokumentationen der Ergebnisse von „*serial recall*“-Versuchen publizierten und somit als Begründer dieses Verfahrens gelten dürfen (Cowan, Sauls, Elliott, & Moreno, 2001).

Die Untersuchungsergebnisse des „immediate serial recall“ weisen verschiedene Effekte auf, von denen ich die Wichtigsten näher beschreiben werde. Sie zeichnen sich dadurch aus, dass sie einander scheinbar widersprechen, woraus die Probleme aktueller Modelle resultieren, alle Effekte erfolgreich zu erklären.

2.2.1 Der „list-length“-Effekt

Der durchschnittliche Anteil der fehlerfrei wiederholten Listen fällt mit zunehmender Listenlänge sigmoidal ab (Botvinick & Plaut, 2006). Dies bezeichnet man als „*list-length*“-Effekt. Dabei können Listen bis zu einer Länge von 4 Elementen im Durchschnitt noch zu über 80% korrekt wiederholt werden. Je nach Art der präsentierten Elemente fällt dann mit zunehmender Länge die Wiederholungsgenauigkeit innerhalb einer Erhöhung der Listenlänge um 3-6 Elemente auf nahezu Null ab (Cranell & Parrish, 1957).

Verläufe der Wiederholungsgenauigkeiten in Abhängigkeit von der Listenlänge werden in Abbildung 1 verdeutlicht.

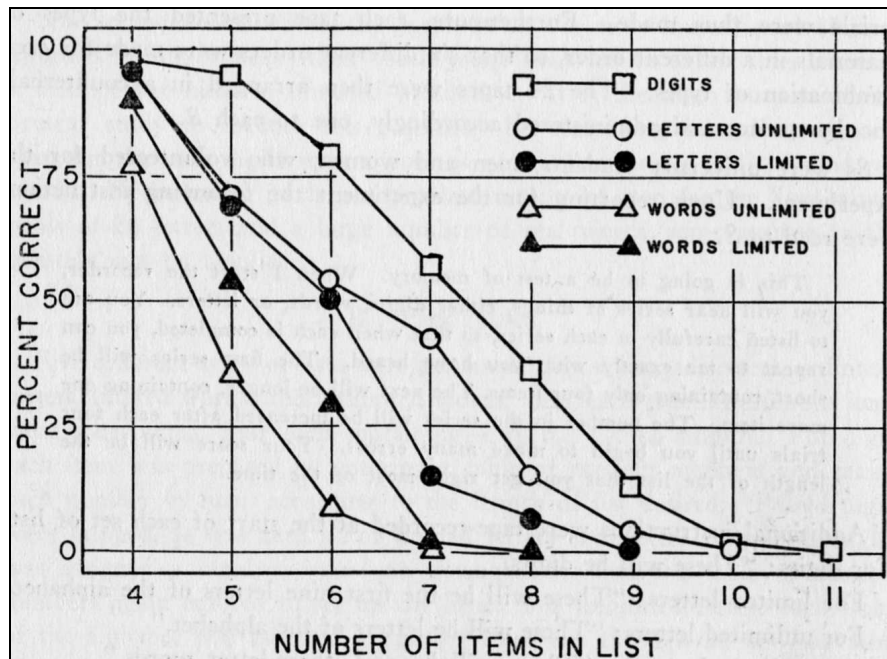


Abbildung 1: Anteil korrekter Wiedergaben von Listen für unterschiedliche Listenlängen (Cranell & Parrish, 1957)

2.2.2 Der „primacy“-Effekt

Der „*primacy*“-Effekt benennt einen durchschnittlich geringeren Fehler bei der Wiederholung der ersten Elemente einer Liste (Jahnke, 1963) als bei der Wiederholung der Elemente, die sich in der Mitte der Liste befinden.

2.2.3 Der „recency“-Effekt

Der „*recency*“-Effekt hingegen beschreibt, dass beim Wiederholen der letzten Elemente einer Liste weniger Fehler gemacht werden, als bei den Elementen aus der Mitte einer Liste (Jahnke, 1963).

2.2.4 „Errors of omission“

Die Fehler, die beim „immediate serial recall“ gemacht werden, lassen sich in mindestens zwei Arten unterscheiden (Conrad, 1959). Eine davon stellen die „*errors of omission*“ dar. Bei einem „error of omission“ wird ein Element der Liste beim Wiederholen ausgelassen. Beispielsweise würde ein solcher Fehler dazu führen, dass die Ziffernfolge 982394 als 98294 wiedergegeben wird. Die Ziffer 3 wurde ausgelassen und die folgenden Ziffern wurden wieder in der korrekten Reihenfolge wiederholt.

2.2.5 „Errors of commission“

Eine zweite Art von Fehler bei der Wiederholung stellen die „*errors of commission*“ oder auch „*substitutions*“ dar.

Beim „error of commission“ wird an Stelle eines Elementes ein Falsches genannt. Ein Beispiel für einen „error of commission“ ist die Wiedergabe der Liste 125294 anstelle von 128294. Hierbei wurde statt der 8 fälschlicherweise eine 5 wiederholt.

In Abbildung 2 kann man die Anteile der unterschiedlichen Fehlerarten bei durchgeführten „serial recall“-Aufgaben erkennen (Schwarz, Reed, Montgomery, Palmer, & Mayer, 1991). Die „Sequence“ und „Substitution“ stellen dabei Fehler der Gruppe der „errors of commission“ dar und „Omissions“ „errors of omissions“. Man kann erkennen, dass mit zunehmender Fehleranzahl die „errors of omissions“ deutlich häufiger auftreten als „errors of commissions“.

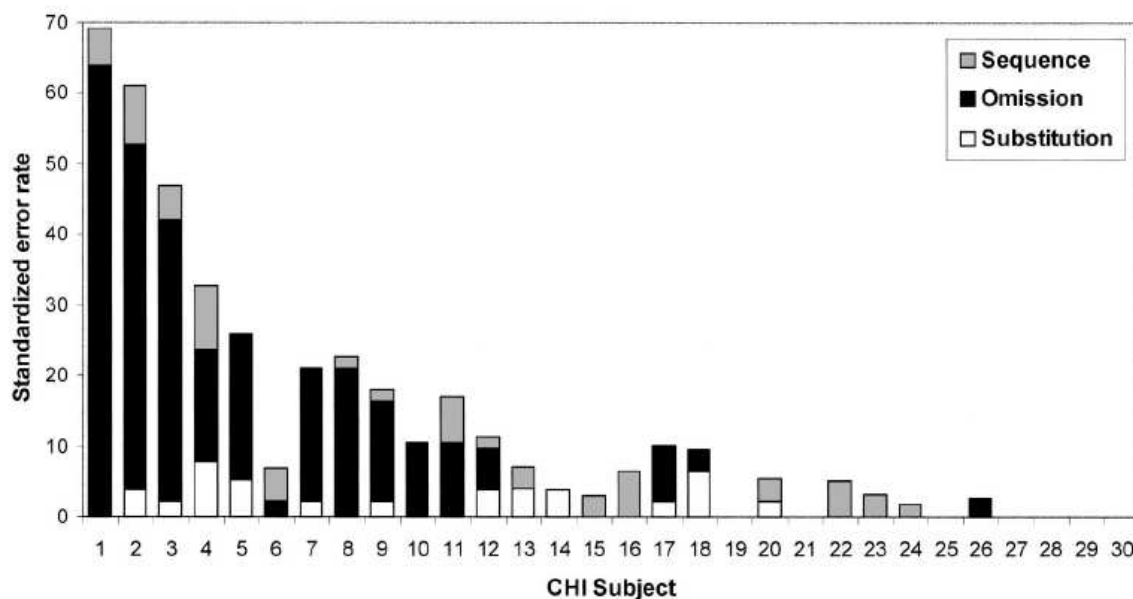


Abbildung 2: Verteilung der unterschiedlichen Fehlerarten bei Patienten mit geschlossenen Kopfverletzungen (Schwarz, Reed, Montgomery, Palmer, & Mayer, 1991)

2.3 Zusammenfassung

Seit den Anfängen der Erforschung des Gedächtnisses wurden viele verschiedene Untersuchungsmethoden entwickelt, um allgemeine oder auch spezielle Eigenschaften offen zu legen. Der „immediate serial recall“ bietet dabei eine verbreitete Untersuchungsmethode die auf Grund ihrer einfachen Durchführung mit zuverlässiger Reproduzierbarkeit diverse Eigenschaften des Arbeitsgedächtnisses aufzeigt. Die zugrunde liegenden Prozesse dieser Effekte sind bis heute nicht bekannt und werden in der aktuellen Forschung auf unterschiedlichste Weise gedeutet (Botvinick & Plaut, 2006). Im Folgenden werde ich eine Auswahl dieser aktuellen Modelle näher beschreiben und deren Erklärungen der Effekte darstellen.

3 Das Arbeitsgedächtnis und verschiedene aktuelle Modelle

Wie im vorherigen Kapitel dargelegt, wurden seit Nipher und Ebbinghaus viele verschiedene Testmethoden entwickelt, um die Phänomene des menschlichen Gedächtnisses aufzuzeigen. Zur Erklärung und Deutung dieser Phänomene wurden parallel verschiedene Modelle entwickelt, die Struktur und Funktionsweise des Gedächtnisses darstellen sollten. Hierfür wurde ein erster Ansatz durch die Unterscheidung in ein Langzeit- sowie ein Kurzzeitgedächtnis anhand der Lernkurve von Ebbinghaus (1885) geschaffen.

Die Tatsache, dass das Gehirn nicht als reiner Datenspeicher funktioniert sondern in der Lage ist, komplexe Aufgaben erfolgreich zu lösen, veranlasste Baddeley und Hitch in ihrer Arbeit „Working Memory“ (1974) dazu, den Begriff des Kurzzeitgedächtnisses durch den des Arbeitsgedächtnisses zu ersetzen (Baddeley & Hitch, 1974).

Bei der Entwicklung von Modellen dieses Arbeitsgedächtnisses wurde in den meisten Fällen das Langzeitgedächtnis separat betrachtet, das wiederum über Schnittstellen mit dem Arbeitsgedächtnis verbunden ist. Unterschiedliche Modelle entstanden zum einen durch verschiedenartige Herangehensweisen aber auch durch verschiedene Schwerpunkte der Wissenschaftler. Manche, wie zum Beispiel Baddeley und Hitch (1974) bei der Entwicklung ihres „Multiple-Component-Model“, versuchten bestimmte empirische Forschungsergebnisse zu erklären während andere, wie Rumelhart (1986), die biologische Funktionsweise der einzelnen Nervenzellen und somit eines Verbundes in den Vordergrund stellten. Wieder andere, so auch Baddeley, Cowan oder Ericsson, unterstützen ihre Aussagen durch Aufnahmen bildgebender Verfahren die Gehirnaktivitäten visualisieren, wie die *Elektroenzephalografie* oder *Magnetresonanztomographie*, mit denen speziellen Arealen bestimmte Funktionen zugeordnet werden können (Baddeley & Logie, 1999; Cowan, 1995; Ericsson & Delaney, 1999). Zunehmende Kapazitäten durch immer leistungsfähigere Computer erlaubten die Implementierung so genannter kognitiver Architekturen wie *EPIC* von Kieras und Meyer (1994) und *SOAR* von Newell (1990), mit deren Hilfe computergestützt kognitive Abläufe modelliert werden können. Da die grundlegenden Mechanismen bereits fertig implementiert sind, können sie nach dem Prinzip eines Baukastens zusammengefügt werden. Hier sind speziell *konnektionistische* Modelle interes-

sant. Diese bestehen aus vielen vernetzten einfachen Einheiten die Informationen über die einzelnen Zustände der Einheiten verteilt repräsentieren. Dadurch stellen sie eine gute Möglichkeit dar, die ähnlichen Strukturen des Gehirns nachzubilden.

Ein großes Problem bei der Vergleichbarkeit aller Modelle stellt die Tatsache dar, dass neben den unterschiedlichen Herangehensweisen der Begriff des Arbeitsgedächtnisses von vielen Forschern unterschiedlich definiert wird.

Im Folgenden werde ich vier aktuelle Modelle des Arbeitsgedächtnisses vorstellen, wobei es sich bei dem zuletzt dargestellten von Botvinick und Plaut um ein computationales Modell handelt. Die ersten drei von Baddeley und Logie, Cowan sowie Ericsson und Delaney sind theoretische Modelle die versuchen, möglichst viele Eigenschaften und Funktionen des Arbeitsgedächtnisses zu beschreiben. Zusätzlich stellen sie sehr einflussreiche Theorien dar, die eine breite Akzeptanz genießen (Miyake & Shah, 1999). Der folgende Überblick wird die Kernthesen der Modelle in einer zusammengefassten Form aufzeigen und vergleichbar machen. Auf diese Weise werde ich versuchen eine Basis zu schaffen, auf der die Probleme und Widersprüche der aktuellen Ansichten deutlich werden.

3.1 Das “Multiple-Component Model” von Alan D. Baddeley und Robert H. Logie

Baddeley und Logie definieren das Arbeitsgedächtnis als die Anzahl an Komponenten die es dem Menschen ermöglichen, ihre aktuelle Umgebung zu repräsentieren, die unmittelbare Vergangenheit zu speichern, die Entwicklung neuen Wissens voranzutreiben, Probleme zu lösen und aktuellen persönlichen Zielen entsprechend zu handeln (Baddeley & Logie, 1999). Das „*Multi-Component-Model*“ des Arbeitsgedächtnisses von Baddeley und Logie stellt eine Weiterentwicklung des gleichnamigen Modells von Baddeley und Hitch von 1974 dar (Baddeley & Hitch, 1974). Hauptmerkmal des Modells ist die Aufteilung des Arbeitsgedächtnisses in spezialisierte Subsysteme, die es dem Menschen durch ihr Zusammenspiel ermöglichen, sich Informationen für kurze Zeit zu merken und sie zu verarbeiten. Im ursprünglichen Modell von 1974 waren diese Subsysteme die „*central executive*“, der „*phonological loop*“ und das „*visuospatial sketchpad*“, die entsprechend der Abbildung 3 miteinander verknüpft waren.

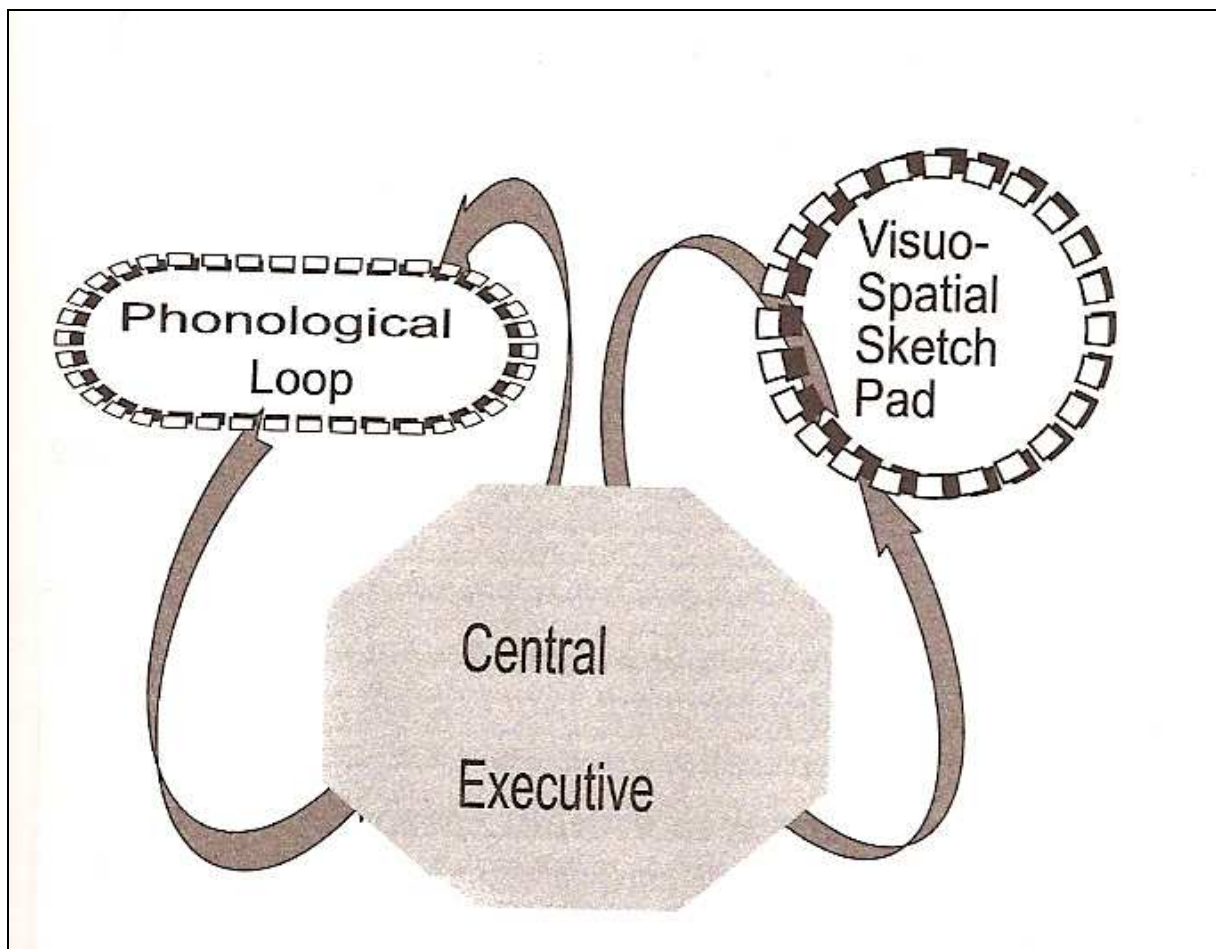


Abbildung 3: Modell von Baddeley und Hitch von 1974 (Baddeley & Logie, 1999)

Die „central executive“ stellte schon damals eine Kontrollinstanz dar, die für die Steuerung der Verarbeitungsprozesse verantwortlich ist. Dabei überwacht und steuert sie, welche Informationen in den anderen beiden so genannten „*slave*“-Systemen gespeichert werden. Das „slave“-System „phonological loop“ kann mit einem Tonband verglichen werden, das mit begrenzter Länge phonetische Daten in einer Schleife immer wieder abspielt und so im Arbeitsgedächtnis bewahrt. Das „visuospatial sketchpad“ ist die Subkomponente, die für die Speicherung von räumlichen und visuellen Informationen genutzt wird. Dort kann, vergleichbar mit einem Notizblock, ein Abbild von gesehenen oder vorgestellten Daten erzeugt und gespeichert werden (Baddeley & Hitch, 1974; Baddeley, 2000).

Dieses „sketchpad“ wurde von Logie (1995) in zwei Subkomponenten aufgeteilt. Anhand der Tatsache, dass Individuen in der Lage sind sich einen räumlichen Ablauf zu merken, wie zum Beispiel den Weg durch einen dunklen Raum, folgerte er, dass auch für den räumlich-visuellen Speicher eine Instanz bestehen muss, die Episoden innerlich abspielen kann.

Spätere Untersuchungen wurden durch die vollständige Trennung in die zwei Subsysteme vor Erklärungsprobleme gestellt, da Patienten in der Lage waren kurze Sequenzen zu wiederholen, obwohl die Funktion des „phonological loop“ unterdrückt wurde. Daraufhin ergänzte Baddeley (2000) sein Modell um den „*episodic buffer*“. Dieser stellt ein drittes „slave“-System dar, welches durch Kontrolle der „central executive“ in der Lage ist, Abfolgen von Daten unterschiedlichen Typs zu speichern. Abbildung 4 zeigt die aktuelle Struktur des Modells.

Die einzelnen Komponenten und ihre Funktionen werden im Folgenden näher beschrieben.

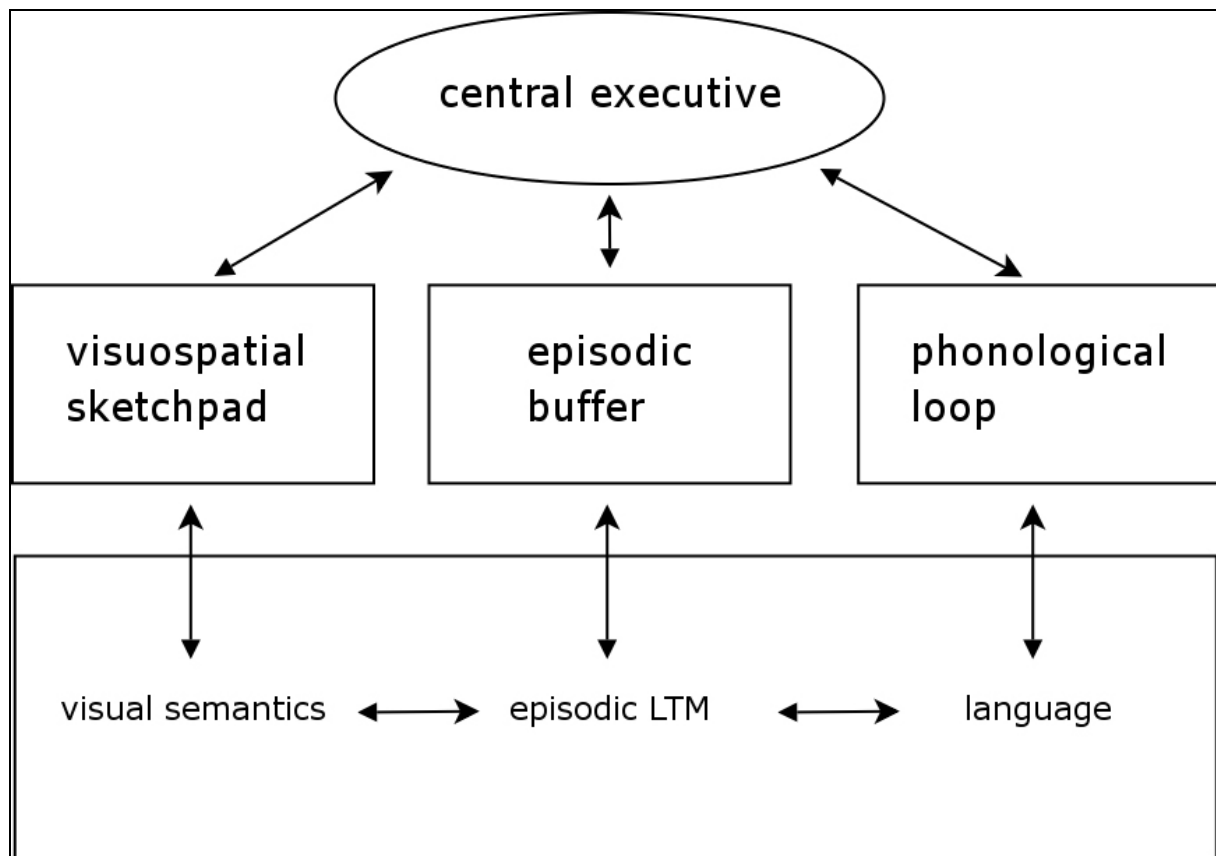


Abbildung 4: Arbeitsgedächtnismodell mit „episodic buffer“ (Baddeley, 2000)

3.1.1 Der „phonological loop“

Baddeley und Logie trennen den „phonological loop“ in einen passiven *akustischen Speicher* und einen aktiven *Rehearsal-Prozess*. Im akustischen Speicher liegen Daten in akustischer Codierung vor und verblassen mit der Zeit. Durch Rehearsal werden die Repräsentationen aufgefrischt, um dem Verblassen entgegenzuwirken (Baddeley & Logie, 1999). Diese Annahme belegen sie mit neuropsychologischen Versuchsergebnissen. So argumentiert Baddeley (1986), dass der Effekt, welcher bei längeren Wörtern größere Fehlerzahlen beim Merken und Wiederholen bewirkt, auf einem Rehearsal-Prozess beruht. Der Effekt hingegen, dass akustisch ähnliche Wörter verwechselt werden, basierend auf den Mechanismen eines passiven akustischen Speichers. Bildgebende Verfahren stützen diese Theorie (Smith & Jonides, 1997).

Um Informationen im Rehearsal-Prozess zu wiederholen werden Mechanismen genutzt, die dem Wahrnehmen von akustischen Daten ähneln. Hinter diesen Mechanismen vermuten Baddeley und Logie dieselben, die auch zur Sprachauffassung und zur Formulierung genutzt werden (Baddeley & Logie, 1999).

Wie auch das andere „slave system“ der „visuospatial sketchpad“ besitzt der „phonological loop“ unterschiedliche individuelle Beschränkungen. Dieses gilt sowohl für die Dauer, die eine Information gespeichert werden kann, als auch für die Menge an Daten, die auf diese Weise aktiviert sein können.

Als Gründe für individuelle Unterschiede nennen Baddeley und Logie (1999) zum einen genetische Faktoren und zum anderen die Folgen von Hirnschädigungen. Es gibt demzufolge Patienten mit beschädigtem verbalem Kurzzeitgedächtnis, die zwar Wörter und Sätze korrekt verstehen können, aber die Repräsentationen nur für kurze Zeit im Gedächtnis behalten können. Hier sehen Baddeley und Logie lediglich eine Schädigung des verbalen Speichers, da diese Patienten ansonsten normale Leistungen bei der Sprachproduktion zeigten (Vallar & Baddeley, 1984). Da hierbei die Funktion von Rehearsal involviert ist, schließen sie eine Schädigung dieses Prozesses aus. Sie erwähnen aber auch Testpersonen, die die Fähigkeiten der Sprachproduktion verloren haben und dadurch Schwierigkeiten beim Rehearsal haben (Caplan, Rochon, & Waters, 1992).

Starke Hinweise auf die Separierungen entsprechend des Modells von Baddeley und Logie zeigen Hirnschädigungen, die eine Fähigkeit intakt lassen, eine andere aber beeinträchtigen. Dabei existieren sowohl Patienten bei denen Fähigkeit A intakt und Fähigkeit B geschädigt ist, als auch Patienten bei denen das Gegenteil der Fall ist. Durch diese so genannte *doppelte Dissoziation* zeigt sich laut Baddeley und Logie, dass verschiedene Systeme für die Fähigkeiten zuständig sind (Baddeley & Logie, 1999).

Bei ihren Untersuchungen zu den kapazitiven Beschränkungen des „phonological loop“ findet der Wortlängeneffekt besondere Beachtung. Hierbei zeigt sich bei der Aufgabe, Wortsequenzen mit kurzen Wörtern zu wiederholen eine geringere Fehleranzahl als bei der selben Aufgabe mit langen Wörtern. Dabei kommt es nicht auf die Anzahl der Buchstaben an, sondern auf die Zeitdauer, die zur Artikulation eines Wortes benötigt wird. Baddeley und Logie (1999) interpretieren aus diesem Phänomen ein zeitliches Verblässen der Repräsentationen als Ursache. Demzufolge würden längere Wörter eine längere Zeit beim Rehearsal in Anspruch nehmen und so anderen Aktivierungen im Arbeitsgedächtnis einen größeren Zeitraum geben, zu verblässen.

Sie schließen auch einen Einfluss von stärkerer Interferenz, hervorgerufen durch längere Worte, wie es von Cowan, Wood, Nugent und Treisman (1997) beschrieben wurde, nicht aus.

3.1.2 Das „visuospatial sketchpad“

Ähnlich wie beim „phonological loop“ wird das „visuospatial sketchpad“ im Modell von Baddeley und Logie in zwei separate Funktionen unterteilt, den passiven *visuellen Puffer* und den aktiven *inneren Schreiber* nach Logie (1995). Hierbei werden die Kapazitäten und Mechanismen für die Speicherung von visuellen Mustern im visuellen Puffer und von Bewegungsabläufen im inneren Schreiber voneinander unterschieden. Anlass dafür gaben Studienergebnisse die zeigten, dass parallel ausgeführte Bewegungen die Speicherfähigkeiten für räumliche Muster verringerten, während irrelevante Bilder die Speicherfähigkeiten für visuelle Informationen verringerten.

Logie und Marchetti (1991) unterstrichen dieses Ergebnis indem sie zeigten, dass zwar räumliche Muster durch Armbewegungen gestört wurden, nicht aber visuelle Informationen wie Farbschattierungen. Auf der anderen Seite konnten solche visuellen Informationen durch irrelevante Bilder gestört werden, nicht aber räumliche Muster.

Diese Unterteilung konnte zusätzlich durch abweichende Entwicklungsgeschwindigkeiten bei heranwachsenden Kindern für die verschiedenen Aufgabenfelder unterstrichen werden (Logie & Pearson, 1997).

Bei der Beurteilung der Begrenzungen der Leistungsfähigkeiten des Subsystems sind laut Baddeley und Logie mehrere Aspekte wichtig. So zeigten Cornoldi, Cortesi und Preti (1991), dass die Anzahl der Dimensionen für eine Orientierungsaufgabe sich stark auf die Performance der Testpersonen auswirkte. So wurde eine Wegfindung in einer 3*3*3 Matrix als deutlich schwieriger empfunden als in einer 5*5 Matrix. Da sich die Anzahl der Felder von 27 und 25 nur geringfügig unterschied, begründeten sie dies mit der zusätzlichen Dimension.

Andere Studien offenbarten, dass bei Unterdrückung von verbaler Codierung optisch ähnliche Gegenstände weniger erfolgreich gemerkt werden können, als deutlich von einander unterscheidbare Objekte. Diese Effekte generierten die Idee, dass sich bei visuellen Daten, ähnlich den akustischen Daten, die Ähnlichkeit störend auf eine erfolgreiche Speicherung auswirkt.

Betreffend der Anzahl an visuellen Objekten erwägen Baddeley und Logie auf Grund von Versuchen, die einen recency-Effekt für genau ein einziges Muster auf-

zeigen, dass auch nur ein Muster zu einem Zeitpunkt im visuellen Gedächtnis gehalten werden kann.

3.1.3 Der „episodic buffer“

Bei einigen dokumentierten Effekten ergaben sich Probleme bei dem Versuch, sie mit dem ursprünglichen Modell zu erklären. Beispielsweise wirkte sich eine visuelle Ähnlichkeit auf die Performance hinsichtlich der Merkfähigkeit von Worten der Testpersonen aus, sich Sequenzen von Worten zu merken. Dies impliziert nach Baddeleys Ansicht eine Verbindung von akustischen und visuellen Daten im Arbeitsgedächtnis (Baddeley, 2000). Weiterhin konnten sich Probanden trotz artikulatorischer Unterdrückung Ziffernfolgen nur mit geringen Leistungseinbußen merken. Theoretisch hätte die Unterdrückung die Leistung des „phonological loop“ viel deutlicher eliminieren müssen. Solche Resultate deuten laut Baddeley (2000) auf einen zusätzlichen Speicher im Arbeitsgedächtnis hin, der Abfolgen von visuellen, akustischen und möglicherweise anderen Informationsarten unterstützt.

Weiterhin werfen die von Miller (1956) erwähnten Datenblöcke (*chunks*) die Frage auf, wo diese Blockabfolgen gespeichert werden, da sie nur selten rein akustische Informationen beinhalten. Gegen die Annahme, dass sie im Langzeitgedächtnis abgelegt werden spricht nach Baddeleys Auffassung die Tatsache, dass sich Patienten mit Amnesien an Textabschnitte nach einer zeitlichen Verzögerung von ungefähr 20 Minuten nicht mehr erinnern können, hingegen bei unmittelbarer Abfrage normale Werte erzielen. Patienten mit dieser Art von Gehirnschädigungen zeigten die Fähigkeit, bei einem Bridge-Spiel die Übersicht zu wahren und sich die gespielten Karten zu merken. Dieses Ergebnis liefert auch einen Hinweis auf eine Speichereinheit für nicht akustische Daten außerhalb des Langzeitgedächtnisses (Baddeley, 2000).

So formuliert Baddeley (2000) den „episodic buffer“ als zusätzliche Komponente des Arbeitsgedächtnisses. Dieser bietet die Möglichkeit verschieden kodierte Informationen in Form einer zeitlichen Repräsentation miteinander zu verbinden. Um diese Aufgabe erledigen zu können, sind die Daten im „episodic buffer“ in einer mehrdimensionalen Kodierung hinterlegt. Hierfür besitzt er nur eine begrenzte Kapazität.

Das Speichersystem für Episoden wird von der „central executive“ kontrolliert. Diese kann durch Steuerung der Aufmerksamkeit auf die Daten zugreifen und durch Lenkung der Aufmerksamkeit Einfluss auf die Inhalte nehmen. So können nicht nur bekannte Zusammenhänge im „episodic buffer“ gespeichert werden, sondern auch neue Repräsentationen konstruiert werden.

3.1.4 Die „central executive“

Wie bei dem ursprünglichen Modell von 1974 vertreten Baddeley und Logie die Auffassung, dass eine zentrale Steuereinheit die Vorgänge im Arbeitsgedächtnis kontrolliert. Im Gegensatz zu den damaligen Ansichten betonen Baddeley und Logie jedoch, dass sie der „central executive“ im aktuellen Modell, neben den Kontrollmechanismen, keine zusätzlichen Speicherfähigkeiten mehr zusprechen (Baddeley & Logie, 1999).

In einem Versuch wurden Personen beauftragt, dargestellte Summen aus zwei Zahlen auf Korrektheit zu überprüfen und sich gleichzeitig dargestellte zusammenhanglose Worte zu merken. Während die Überprüfung Mechanismen der Verarbeitung beanspruchen sollte, wurde für die zweite Aufgabe lediglich Speichermechanismen benötigt. Hierbei wurde durch Vergleichen der Ergebnisse für die Doppelaufgabe mit den Testergebnissen der Einzelaufgaben gezeigt, dass die Leistung für die beiden Aufgaben nicht durch simultane Ausführung beeinflusst wird (Duff & Logie, 2001). Baddeley und Logie leiten daraus ab, dass Speicherung und Verarbeitung getrennte Ressourcen nutzen. So erklären sie, dass bei dem erwähnten Versuch die Worte im „phonological loop“ abgelegt werden während die „central executive“ die Verifikation der Additionen übernimmt (Baddeley & Logie, 1999).

Bei Ergebnissen, in denen die sonstigen Kapazitäten des Arbeitsgedächtnisses übertroffen werden, argumentieren sie für einen Einfluss des Langzeitgedächtnisses beziehungsweise anderer Subsysteme. Als Gründe hierfür nennen Baddeley und Logie zum einen, dass sonst eine übermächtige „central executive“ nicht mehr produktiv analysiert werden kann.

Zum anderen hat sich gezeigt, dass „recency“-Effekte sich auf komplexe Strukturen auswirken können, wie beispielsweise ganze Textpassagen. „Recency“-Effekte beschreiben das Phänomen, dass die letzten Elemente einer Sequenz oder auch der letzte Abschnitt eines Textes deutlicher im Gedächtnis erinnert wird (vgl. Kapitel 2.2.3). Dieses äußert sich beispielsweise durch geringere Fehlerzahlen beim Wiederholen der Elemente, die am Ende einer Liste gelernt wurden.

Dieser „recency“-Effekt weist laut Baddeley und Logie auf einen Einfluss des Arbeitsgedächtnisses hin (Baddeley, 2002). Da die „central executive“ als Speicher ausgeschlossen wurde und weder der „phonological loop“ noch das „visuospatial sketchpad“ solch komplexe Strukturen erfassen können, wird laut Baddeley eine zu-

sätzliche Speichermöglichkeiten neben den beiden „slave“-Systemen benötigt, die Baddeley(2000) als den „episodic buffer“ vorstellte (vgl. Kapitel 3.1.3).

Als Aufgabenbereiche der „central executive“ benennen Baddeley und Logie außer der Koordination der „slave systeme“ die Steuerung der Aufmerksamkeit und die Aktivierung von Informationen aus dem Langzeitgedächtnis. Sie lassen dabei aber offen, ob es sich wirklich um ein einheitliches Konstrukt handelt und in welcher Form dieses Steuerungsorgan strukturiert ist (Baddeley, 1996).

Im Bezug auf Aufmerksamkeit und Bewusstsein weisen Baddeley und Logie der „central excutive“ eine wichtige Rolle zu. Sie sehen Aufmerksamkeit und Arbeitsgedächtnis zwar eng miteinander verbunden, distanzieren sich aber davon, die Begriffe gleichzusetzen. So bezieht sich der Begriff der Aufmerksamkeit laut Baddeley und Logie auf Kontrollmechanismen während der Begriff Arbeitsgedächtnis die Speicherfunktionalität widerspiegelt (Baddeley & Logie, 1999).

Im Hinblick auf das Bewusstsein erwähnen sie eine Studie von Teasdale, Dritschel, Taylor, Proctor, Lloyd, Nimmo-Smith und Baddeley (1995), in welcher der Einfluss der zentralen Steuereinheit bei der Generation irrelevanter Bilder, die keinen Zusammenhang zur aktuellen Tätigkeit besitzen zu scheinen, nachgewiesen wird. Dadurch wird ihrer Ansicht nach verdeutlicht, dass die zentrale Steuereinheit zum aktuellen Bewusstseinsbild beiträgt (Baddeley, Logie, 1999).

3.1.5 Zusammenspiel der Komponenten bei kognitiven Aufgaben

Die vorher beschriebenen Komponenten des Modells tragen durch ihr koordiniertes Zusammenspiel auf verschiedene Art zur Lösung von Arbeitsgedächtnisaufgaben bei.

Baddeley und Logie sind zwar davon überzeugt, dass beim Sprachverstehen der „phonological loop“ eine wichtige Rolle spielt, zeigen aber anhand von verschiedenen empirischen Testresultaten, dass noch weitere unabhängige Komponenten beteiligt sein müssen. Wäre ausschließlich der „phonological loop“ involviert, würden Techniken wie die artikulatorische Unterdrückung, die dessen Funktionen stören, die Fähigkeit zum Sprachverstehen erheblich einschränken. Solche Techniken zeigen zwar Effekte in Hinsicht auf Leseaufgaben, aber auf das Verstehen an sich haben sie nur geringen Einfluss (Baddeley & Logie, 1999). Andere Studien weisen darauf hin, dass das „visuospatial sketchpad“ beim Prozess des Lesens einbezogen wird. Brooks (1967) zeigte zum Beispiel, dass Lesen das Merken von Bildern behindert. Eine weitere grundlegende Rolle beim Verständnis spielt laut Baddeley und Logie die „central

executive“ (Gathercole & Baddeley 1993). Sie konstruiert eine innere Repräsentation der Zusammenhänge und aktiviert dann die zugehörigen Informationen im Langzeitgedächtnis. Somit ist es für das erfolgreiche Verständnis einerseits von Bedeutung ob es Strukturen im Langzeitgedächtnis gibt, die die Zusammenhänge erklären können und andererseits, wie erfolgreich die „central executive“ dabei eine flexible aber klare Repräsentation im Arbeitsgedächtnis zu erzeugen vermag.

Amnestische Patienten können demzufolge eine korrekte Repräsentation im Arbeitsgedächtnis erzeugen, haben aber Probleme diese im Langzeitgedächtnis zu speichern. Alzheimer Patienten hätten dann bereits ein Problem bei der vorangehenden Erstellung der Repräsentation (Baddeley & Logie, 1999).

Ein anderes Beispiel für die Zusammenarbeit der Subkomponenten sind die unterschiedlichen Anforderungen, die beim Spielen eines Videospiele an das Arbeitsgedächtnis gestellt werden. Hierbei verglichen Baddeley und Logie (1999) die Leistungen und deren Störanfälligkeiten sowohl nach einer Einspielphase als auch nach einem folgendem längerem Training.

Anfangs wurde die Performance der Testpersonen durch parallele verbale als auch räumlich-optische Aufgaben stark beeinflusst. Eine gleichzeitige motorische Aufgabe, wie das Tippen mit dem Fuß hatte aber nur sehr geringe Auswirkungen. Nach einer längeren Trainingszeit kehrte sich dieser Einfluss um. Die verbalen und räumlich-optischen Aufgaben zeigten nur geringe Effekte und das Tippen mit dem Fuß hatte eine größere Störwirkung. Dieses Phänomen erklären Baddeley und Logie (1999) damit, dass zu Beginn die Menge der Informationen die vom Arbeitsgedächtnis zur Steuerung benötigt wurden überwogen und der Bewegungsapparat nur wenig beansprucht wurde.

Mit zunehmender Routine wurden dann spezialisierte Fähigkeiten im Bewegungsapparat entwickelt die dazu führten, dass weniger Kapazität vom generellen Speichervermögen des Arbeitsgedächtnisses genutzt werden musste (Logie, Baddeley, Mane, Donchin, & Sheptak, 1989).

Mit diesen und ähnlichen Versuchsergebnissen für andere Arbeitsgedächtnisaufgaben stützen Baddeley und Logie ihre Theorie, die das Arbeitsgedächtnis in mehrere Teilkomponenten einteilt, die bei verschiedenen Aufgaben zu unterschiedlichen Anteilen involviert sind.

3.1.6 Langzeitgedächtnis und Arbeitsgedächtnis

Baddeley und Logie (1999) betonen, dass trotz der Integration von Langzeitgedächtnisinhalten in Arbeitsgedächtnisprozesse durch die „central executive“ diese als voneinander getrennte Systeme betrachtet werden müssen.

Im Bezug auf den „phonological loop“ deuten viele Hinweise auf einen großen Einfluss des Langzeitgedächtnisses auf Arbeitsgedächtnisprozesse hin. So können längere Sequenzen von Buchstaben oder Wörtern im Gedächtnis abgelegt werden, wenn sie offensichtliche semantische oder syntaktische Beziehungen aufweisen. Sie werden dann entsprechend der Theorie von Miller (1956) zu Blöcken (*chunks*) zusammengefasst. Diese Blöcke erfordern weniger Kapazität im Arbeitsgedächtnis, da sie auf Kapazitäten im Langzeitgedächtnis zurückgreifen.

Auf der anderen Seite zeigen auch die Mechanismen des „phonological loop“ eine Einflussnahme auf die Entwicklung von Strukturen im Langzeitgedächtnis, wie zum Beispiel beim Spracherwerb. So behindert ein beschädigter oder unterdrückter „phonological loop“ Testpersonen bei der Bildung neuer Repräsentationen von akustischen Daten im Langzeitspeicher (Baddeley & Logie, 1999).

Diese Doppelbeziehung kann in den Augen von Baddeley und Logie nicht in einem Modell erfasst werden, welches das Arbeitsgedächtnis als bloße Aktivierung von Daten im Langzeitspeicher beschreibt, wie es beim „Embedded-Process Model“ von Cowan (vgl. Kapitel 3.2) der Fall ist.

Laut Baddeley und Logie würden in einem derartigen Modell neue akustische Daten zu einer Aktivierung einer vorhandenen Repräsentation führen, die diesen Daten am ehesten entsprechen. Dadurch würden dann neue Daten als bekannte Fakten fehlinterpretiert werden. Auf diese Weise könne nicht beschrieben werden, wie neue akustische Daten gelernt werden können.

Weiterhin zeigten Versuchspersonen mit geschädigtem Kurzzeitgedächtnis nur geringe bis keine Probleme beim Sprachverständnis (Shallice & Warrington, 1970) sowie Versuchspersonen mit gestörtem Sprachverständnis teilweise normale Kurzzeitgedächtniskapazität (Baddeley & Wilson, 1993). Wäre der „phonological loop“ also lediglich die Aktivierung von Langzeitwissen, würde ein beschädigtes Kurzzeitgedächtnis zu einer beeinträchtigten Aktivierung und somit zu einer Beeinträchtigung der Sprachauffassung führen.

Für das „slave system“ des „visuospatial sketchpad“ gilt diese Separierung entsprechend. Auch hier gelten Beziehungen zwischen Langzeitgedächtnis und den

räumlich-visuellen Daten des Arbeitsgedächtnisses als gemeinhin akzeptiert. Trotzdem sprechen Untersuchungen für die Unterteilung in zwei Systeme, wobei die aktivierten Daten aus dem Langzeitgedächtnis jeweils in das entsprechende Subsystem des Arbeitsgedächtnisses transferiert werden. Untersuchungen mit Patienten, die an einem *visuellen Neglect* leiden, unterstützen diese These (Baddeley, 1999). Sie zeichnen sich unter anderem dadurch aus, dass sie eine Seite ihrer Umgebung, von Objekten und von ihrem Körper ignorieren (Bisiach, 1993). Wenn sie einen bekannten Ort oder Raum aus dem Gedächtnis beschreiben sollen, lassen sie eine Hälfte des Ortes außer Acht. Wenn sie sich ihren eigenen Standpunkt aber auf der entgegen gesetzten Seite vorstellen, wird die andere Seite ignoriert und die vorher fehlende beschrieben. Dieses weist auf eine intakte Repräsentation im Langzeitgedächtnis hin, die aber im Arbeitsgedächtnis nur noch teilweise erkannt wird.

3.1.7 Bewertung des „Multiple-Component-Model“

Das Arbeitsgedächtnis besteht nach Ansicht von Baddeley und Logie aus mehreren spezialisierten Teilkomponenten, die durch ihre Zusammenarbeit dem Menschen ermöglichen, seine aktuelle Situation zu begreifen und aus seinen Erfahrungen zu lernen sowie Probleme zu lösen.

Das Modell wurde aus Studien sowohl mit gesunden als auch mit hirngeschädigten Testpersonen unterschiedlichen Alters und einer breiten Auswahl an unterschiedlichen Experimenten entwickelt.

Mit diesem Modell können viele kognitive Aufgaben abgebildet werden und es lässt sich eine breite Palette von empirischen Daten damit erklären. Somit konnte es sich seit der Einführung vor über 30 Jahren mit nur wenigen Aktualisierungen und Ergänzungen als Rahmengebilde mit einer großen Akzeptanz durchsetzen.

Diese Flexibilität scheint darauf zu beruhen, dass viele Aspekte des Modells, wie beispielsweise die genauen Mechanismen der „central executive“ sehr unklar definiert wurden und damit genügend Freiraum für verschiedene Interpretationen offen ließ.

Die biologische Plausibilität des Modells wird auch nur insofern deutlich, als die Aktivität verschiedener Areale im Gehirn mit der angenommenen Inanspruchnahme der verschiedenen Teilkomponenten korreliert.

Meines Erachtens konnte sich das „Multiple-Component-Model“ von Baddeley und Logie so beharrlich behaupten, da es nur wenige klare Aussagen beinhaltet. Nach der Einführung des „episodic buffer“ wurde die vorherige klare Trennung von

„visuospatial sketchpad“ und „phonological loop“ zum Teil wieder aufgehoben. Genauere Mechanismen und Abgrenzungen der „central executive“ werden ebenfalls nicht näher erläutert.

Somit stellt das Modell meiner Ansicht nach nur eine vage Beschreibung von beobachteten Verhalten dar und kann letztendlich keine konkreten Erklärungen für Verhaltensweisen liefern.

Ebenso vage beschreibt das Modell die Mechanismen beim „immediate serial recall“. Für die Aufgabe des „immediate serial recall“ spielen bei verbal kodierten Elementen die Kapazitäten des „phonological loop“ eine bedeutende Rolle (Baddeley, 1999, 2000, 2002). Sollten die einzelnen Listenelemente visuell kodiert werden oder eine Kombination aus verbaler und visueller Kodierung bilden, trägt der „episodic buffer“ zur Verknüpfung der verbalen und visuellen Informationen bei (Baddeley, 2000, 2002). Da sowohl „phonological loop“ als auch „episodic buffer“ eine feste Beschränkung besitzen, können nur Listen bis zu einer bestimmten Länge gespeichert werden (vgl. Kapitel 3.1.1 und 3.1.3). Somit scheint das Modell den „list length“-Effekt (vgl. Kapitel 2.2.1) erklären zu können.

Die Repräsentationen, die in „phonological loop“ und „episodic buffer“ gespeichert sind, werden durch den Rehearsal-Prozess aufrechterhalten (Baddeley, 1986, 1999, 2000, 2002). Elemente, die am Anfang der Liste stehen, befinden sich länger im Arbeitsgedächtnis und erfahren somit eine größere Anzahl an Rehearsals. Dadurch werden diese Listenelemente stärker repräsentiert, wodurch der „primacy“-Effekt beim „immediate serial recall“ erklärt werden kann.

Eine Erklärung für den „recency“-Effekt sowie für die „errors of omission“ und die „errors of commission“ kann das Modell von Baddeley und Logie meines Erachtens aber nicht liefern. Hierfür fehlen explizite Angaben darüber, in welcher Form die Daten im Arbeitsgedächtnis gespeichert werden sowie eine genaue Beschreibung der Mechanismen, die durch die Speicherung und anschließende Abrufung der Daten bewirkt werden. Baddeley und Hitsch (1993) beschreiben die grundlegenden Mechanismen des „recency“-Effektes zwar mit einer passiven Kurzzeitspeicherung, ähnlich der Prozesse beim *Priming* oder beim Aneignen *impliziten Wissens*, formulieren solche Mechanismen aber nicht in der Beschreibung ihres Modells. Beim *Priming* bewirkt beispielsweise die vorherige Präsentation eines Objektes, dass es bei einer späteren Aufgabe schneller assoziiert wird. Implizites Wissen stellt angeeignete Fähigkeiten dar, wie zum Beispiel bestimmte Bewegungen, während im Gegensatz da-

zu explizites Wissen zum Beispiel Daten und Zusammenhänge darstellt, die im Gedächtnis repräsentiert sind. Diesen Zusammenhang zwischen „recency“-Effekten und implizitem Wissen bzw. Priming führen Baddeley und Hitsch (1993) auf grundlegende Gemeinsamkeiten zurück. Diese Gemeinsamkeiten basieren auf gleichen Eigenschaften von implizitem Wissen bzw. Priming und dem „recency“-Effekt beispielsweise bei Patienten mit Amnesie in verschiedenen Entwicklungsstadien und unter Drogeneinfluss. Wegen dieser grundlegenden Gemeinsamkeiten schlussfolgern Baddeley und Hitsch (1993), dass die Mechanismen beim „recency“-Effekt denen beim impliziten Wissen bzw. beim Priming entsprechen. Es existiert somit ein genügend großer Freiraum im Arbeitsgedächtnismodell von Baddeley und Logie für eine Erklärung des „recency“-Effektes, welcher allerdings noch mit der expliziten Angabe von Mechanismen gefüllt werden muss.

Bei den Angaben zur Speicherung von Reihenfolgen existiert ebenfalls ein solcher Freiraum. Es wird ausgeschlossen, dass die gespeicherten Daten jeweils mit ihrem Nachfolger verknüpft werden, da ein solches *chaining* durch Untersuchungsergebnisse widerlegt wird (Baddeley, 2002). Dabei machten Versuchspersonen bei der Aufgabe eine Liste zu wiederholen, in der gleichartige Elemente mit sich unterscheidenden Elementen abwechselten, mehr Fehler bei den ähnlichen Elementen, als bei den sich unterscheidenden Elementen, die diesen folgten. Somit wird zwar eine Speichermethode ausgeschlossen, eine Alternative wird aber nicht festgelegt.

Damit kann das Modell mit den beschriebenen Mechanismen sowohl den „list length“-Effekt als auch den „primacy“-Effekt erklären, der „recency“-Effekt sowie die „errors of omission“ und „commission“ werden jedoch nur vom Modell erfasst, da die unpräzisierte Formulierung Platz für mögliche Erklärungen lässt. Das heißt, auch wenn die Flexibilität des Modells dazu führt, dass eine Erklärung des „recency“-Effektes und der zwei beschriebenen Fehlerarten nicht ausgeschlossen werden, fehlen doch jegliche Nachweise, in welcher Weise dies möglich sein könnte.

3.2 Das „Embedded-Process Model“ von Nelson Cowan

Cowan definiert das Arbeitsgedächtnis als die kognitiven Prozesse, die Informationen in leicht zugänglicher Form aufbewahren, um damit beliebige Aufgaben mit gedanklichem Anteil ausführen zu können (Cowan, 1999). Hierzu hat er ein theoretisches Rahmenmodell entworfen, das im Wesentlichen auf der hierarchischen Anordnung von Langzeitgedächtnis, aktivierten Erinnerungen und dem Mittelpunkt der Aufmerksamkeit basiert. Das Langzeitgedächtnis stellt hierbei die Gesamtheit aller gespeicherten Informationen, Prozesse und Zusammenhänge dar. Verschiedene externe und interne Stimuli bewirken die Aktivierung von Teilen dieser Informationen und machen sie dadurch greifbarer für informationsverarbeitende Prozesse. Im aktuellen Mittelpunkt der Aufmerksamkeit befindet sich aber wiederum nur ein kleiner Teil der aktivierten Daten.

Auf diese Weise werden die Daten, die für eine Aufgabe des Arbeitsgedächtnisses benötigt werden, unterschiedlich deutlich bereitgestellt. Eine „central executive“ steuert den Fokus der Aufmerksamkeit und somit die Verarbeitung dieser Daten, so dass als Resultat eine Aktivierung einer Lösungsrepräsentation entsteht.

Das Arbeitsgedächtnis ist also laut Cowan (1988) in das Langzeitgedächtnis eingebettet und beinhaltet alle aktivierten Daten und die Informationen im Fokus der Aufmerksamkeit. Zusätzlich existiert ein sensorischer Speicher, der für wenige Millisekunden Sinnesreize speichert.

Die Abbildung 5 zeigt den schematischen Aufbau dieses Modells sowie Mechanismen zur Bearbeitung und Auswertung eines Reizes durch das Arbeitsgedächtnis.

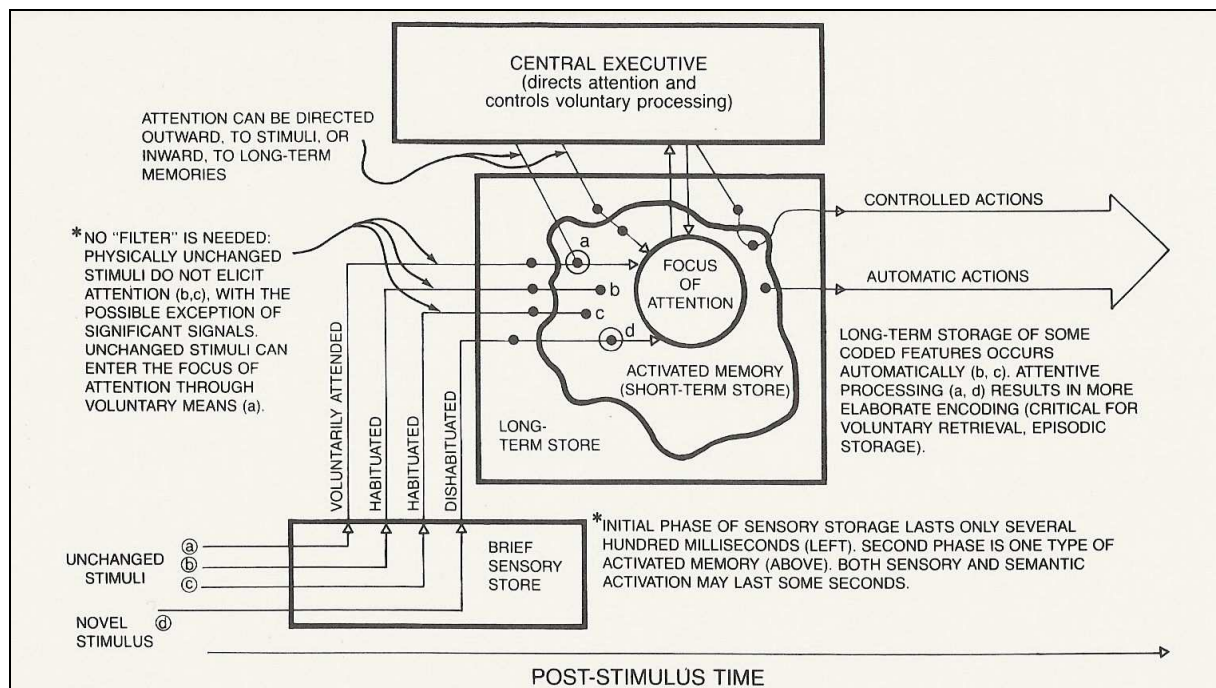


Abbildung 5: Arbeitsgedächtnismodell nach Cowan (Cowan, 1999)

Indem Cowan den Fokus der Aufmerksamkeit als Teilmenge der aktivierten Informationen darstellt verdeutlicht er, dass man Informationen immer aktiviert, wenn man ihnen Beachtung schenkt. Auf der anderen Seite können hingegen Daten des Langzeitgedächtnisses aktiviert werden, ohne in das Zentrum der Aufmerksamkeit zu rücken (Cowan, 1999).

Nach Cowan resultiert jeder Reiz in einer Aktivierung von Gedächtnisinhalten. Dabei sind aber die Eigenschaften der Daten, die in den Fokus der Aufmerksamkeit gelangen im Gegensatz zu den anderen aktivierten Daten stärker und zahlreicher erfasst. So wird beispielsweise durch das Vernehmen des eigenen Namens durch einen unbeachteten Kanal eine Repräsentation im Langzeitgedächtnis aktiviert. Wenn diese Repräsentation umfassend genug ist führt sie dazu, dass die Aufmerksamkeit darauf gelenkt wird. Dadurch werden dann zusätzliche Eigenschaften, die mit der Information verknüpft sind, aktiviert.

Wie in Abbildung 5 gezeigt, geschieht die Zuteilung von Aufmerksamkeit entweder (a) durch Steuerung der zentralen Kontrollstruktur oder (d) automatisch durch besonders herausstechende Stimuli wie laute Geräusche oder helle Lichter.

Zum Arbeitsgedächtnis tragen alle drei Komponenten bei: die Aktivierung, der Fokus der Aufmerksamkeit und die Teile des Langzeitgedächtnisses, die mit aktivierten Daten verknüpft sind. Die Informationen im Fokus sind dabei umfassendsten greifbar, während die Informationen aus der Menge der Aktivierung nur mit etwas größerem

Zeitaufwand zu aktuellen Verarbeitungsprozessen hinzugezogen werden können. Sollten zusätzlich Zusammenhänge aktuell im Gedächtnis aktiviert sein, die mit Erinnerungen aus dem Langzeitgedächtnis verknüpft sind, können auch diese Daten aktiviert und dann ebenfalls in den Fokus der Aufmerksamkeit gerückt werden. Die einzelnen Komponenten erfahren dabei unterschiedliche Beschränkungen auf verschiedene Arten. So nimmt Cowan an, dass der Fokus der Aufmerksamkeit hauptsächlich in Bezug auf die Kapazität begrenzt ist, während die aktuelle Aktivierung von Gedächtnisinhalten durch gegenseitige Interferenzen von ähnlichen Daten und möglicherweise auch durch zeitlichen Verfall der Repräsentation zeitlich begrenzt ist.

Informationen, die sich gleichzeitig im Fokus der Aufmerksamkeit befinden, sind durch diese momentane Repräsentation miteinander verbunden. In gleicher Weise sind die aktivierten Daten dadurch miteinander verbunden, dass sie sich im gleichen Zustand befinden. Diese Verbindungen von Informationen können sich zu neuen Verknüpfungen dieser Informationen im Langzeitgedächtnis entwickeln (Cowan, 1999). So können dann beispielsweise neue Wörter oder Abläufe gelernt werden. Somit ist das Arbeitsgedächtnis nach dem Modell von Cowan in der Lage, neue Wörter zu lernen ohne diese entsprechend der Kritik von Baddeley und Logie als bekannt fehlzuinterpretieren (vgl. Kapitel 3.1.6).

Mit diesem Modell beabsichtigt Cowan die allgemeinen Vorgänge des Arbeitsgedächtnisses abzubilden (Cowan, 1999). Im Gegensatz zu Baddeley und Logie versucht er nicht, Kodierungen oder Speichermechanismen für speziellen Datentypen zu unterscheiden, wie es in deren Modell zwischen „phonological loop“ und „visuospatial sketchpad“ der Fall ist. Stattdessen beschreibt er mit seinem Modell die generellen Mechanismen, die für alle Arten von Informationen gelten sollen (Cowan, 1999).

3.2.1 Mechanismen des Arbeitsgedächtnisses

Als wesentliche Verarbeitungsmechanismen des Arbeitsgedächtnisses führt Cowan die Kodierung, Repräsentation, Erhaltung und Wiedergewinnung von Informationen auf.

Ein Reiz bewirkt durch eine entsprechende Verknüpfung der Sinneszellen, die ihn aufnehmen ein spezielles Aktivierungsmuster von entsprechenden Erinnerungen im Langzeitgedächtnis. Die Kodierung des Reizes erfolgt in Cowans Modell durch das individuelle Muster der Aktivierungen, die durch diesen Reiz hervorgerufen wurde. Wenn ein Reiz nur schwach erfolgt oder auf einem unbeachteten Kanal ausgelöst wird, beinhaltet die Aktivierung jedoch nur einen Teil aller Eigenschaften der Reprä-

sensation. Dies sind laut Cowan (1999) eher physische als semantische Merkmale, die mit dem Reiz verbunden werden. Wenn dem Reiz mehr Aufmerksamkeit zugeordnet wird, kann eine komplexere und mehrdimensionale Repräsentation auf der Basis unterschiedlicher Kodierungen entstehen.

Somit distanziert sich Cowan bezüglich der Repräsentation auch von der Idee, dass Signale eines bestimmten Typs auch nur in dieser Weise gespeichert werden. Das heißt, von akustischen Signalen existieren nicht nur akustische Repräsentationen sondern möglicherweise auch visuelle Abbilder. Je nach aktueller Aufgabenstellung kann es sinnvoll sein, die Information in einer bestimmten Kodierung zu verarbeiten. So kann ein Wort orthographisch, akustisch und semantisch kodiert sein. Für das Merken einer Sequenz kann es vorteilhaft sein, die akustische Repräsentation per Rehearsal im Gedächtnis zu behalten, während für räumliche Vorstellungen eine visuelle Repräsentation geeignet ist.

Die Aufrechterhaltung einer Information im Arbeitsgedächtnis beschreibt Cowan (1999) als die Mechanismen, die dazu beitragen, dass die Aktivierung der entsprechenden Repräsentationen erneuert wird. Baddeley's Theorie vom verbalen Rehearsal (vgl. Kapitel 3.1) setzt er mit dem Behalten von Informationen im Fokus der Aufmerksamkeit gleich. Zusätzlich beschreibt Cowan noch das gesteuerte Durchsuchen von Gedächtnisinhalten. Dabei geraten Daten ebenfalls in den Mittelpunkt der Aufmerksamkeit und können gegebenenfalls so reaktiviert werden. Als Nachweis für diese Theorie führte er die Ergebnisse seiner Studien zu Antwortzeiten von vierjährigen bei der Aufgabe an, sich Wortlisten zu merken (Cowan, 1992). Hierbei zeigte sich, dass die Antwortzeiten bei Kindern, die sich längere Listen merken konnten deutlich länger wurden. Dieses erklärt Cowan durch eine Reaktivierungsphase zwischen den einzelnen Wortausgaben. Dass dieses aber nicht ein verbales Rehearsal darstellt argumentiert Cowan anhand der Tatsache, dass der Wortlängeneffekt für die Dauer der Reaktivierungsphase keine Rolle spielt.

Cowan geht also davon aus, dass beim Aufrechterhalten von Arbeitsgedächtnisinhalten mehrere Mechanismen kombiniert Einfluss nehmen.

Die Wiedergewinnung von Informationen definiert Cowan als das Zurückrufen in den Fokus der Aufmerksamkeit. Beim Zurückholen von Informationen aus der Menge der aktivierten Daten spielt auf Grund der temporären Begrenzung der Aktivierung die Zeit eine wichtige Rolle. Wenn die Aktivierung dieser Daten vollständig abgeklungen ist, können sie aus dem Langzeitgedächtnis nur wieder gewonnen werden, wenn

sie dort in einem ausreichend nachvollziehbaren Zusammenhang abgelegt worden sind.

Auf die Steuerung und Kontrolle dieser Mechanismen werde ich im nächsten Kapitel genauer eingehen.

3.2.2 Steuerung des Arbeitsgedächtnisses

Die Steuerung des Arbeitsgedächtnisses erfolgt über die Kontrolle des Mittelpunktes der Aufmerksamkeit durch die „central executive“ (Cowan, 1999). Da die Daten, die sich im Fokus der Aufmerksamkeit befinden gleichzeitig aktiviert werden, wird so auch die Menge der aktivierten Daten gesteuert. Diese bleiben jedoch noch einige Zeit aktiviert, selbst wenn sie sich nicht mehr im Mittelpunkt der Aufmerksamkeit befinden.

Steuerung erfolgt durch die Separation der Daten die in den Fokus der Aufmerksamkeit gerückt werden, von denen die gegebenenfalls explizit davon ausgeschlossen werden.

Dabei kann diese Selektion einmal automatisch erfolgen oder durch die zentrale Steuereinheit bewusst gesteuert werden.

Testergebnisse haben bestätigt, dass es leichter ist seine Aufmerksamkeit auf einen Kanal zu konzentrieren, wenn sich dieser durch physikalische Eigenschaften von anderen Kanälen unterscheidet, im Gegensatz zu Kanälen, die sich durch ihre semantische Bedeutung unterscheiden. Da es sich aber gezeigt hat, dass bis zu einem gewissen Grad auch die semantischen Informationen bei dieser Selektion berücksichtigt werden, beschreibt Cowan den Mechanismus der Kontrolle entsprechend Solokov's Theorie der „*orienting response*“ (Solokov, 1963). Diese besagt, dass im Gedächtnis eine Repräsentation sämtlicher aktueller Reize gebildet wird. Neu auftretende oder veränderte Reize werden mit dieser internen Repräsentation verglichen. Die Aufmerksamkeit wird nur dann erweckt, wenn der neue Reiz eine Abweichung oder einen Widerspruch zur aktuellen Repräsentation darstellt. Sich wiederholende Reize werden jedoch mit der Zeit in die Repräsentation mit aufgenommen und erlangen somit immer weniger Aufmerksamkeit.

Nach dem Prinzip dieser „*orienting response*“ werden laut Cowan im Arbeitsgedächtnis die Informationen verschiedener Kanäle automatisch gefiltert. Jeder Reiz ruft zwar eine Aktivierung von Gedächtnisinhalten hervor, Aufmerksamkeit bekommt er jedoch nur, wenn die Bedingungen des Filters dieses zulassen.

Um diese Theorie zu belegen erwähnt Cowan (1999) ein Beispiel, bei dem Versuchspersonen mathematische Aufgaben lösen und dabei von einer Audioaufnahme mit mathematischen Ausdrücken, wie „plus“, „gleich“, „drei“ usw. abgelenkt werden. Wenn den Testpersonen die Aufnahme vorher einmal vorgespielt wurde, fiel der Einfluss der Ablenkung weit weniger stark aus, als wenn die Aufnahme beim Test zum ersten Mal gehört wurde.

Wenn einem wiederholten Reiz aber trotzdem Aufmerksam zugeteilt werden soll, so ist dieses in Cowan's Modell nur durch die aktive Zuweisung durch die zentrale Steuereinheit möglich.

Diese passiven und aktiven Aufmerksamkeitssteuerungen beeinflussen sich untereinander. In einem leisen Raum kann einem Redner leichter gefolgt werden, da die Aufmerksamkeit sowohl durch automatische Prozesse als auch gesteuert darauf gelenkt wird, als unter Bedingungen mit einer Vielzahl an Störsignalen, die durch automatische Zuweisung immer wieder in den Fokus der Aufmerksamkeit gelangen.

3.2.3 Beschränkungen des Arbeitsgedächtnisses

Bei Cowan werden zwei mögliche Arten der Beschränkung für das Arbeitsgedächtnis in Betracht gezogen. Zum einen, dass Informationen durch Interferenz mit ähnlichen Daten im Arbeitsgedächtnis überschrieben werden und zum anderen, dass die Menge dieser Informationen zu einem bestimmten Zeitpunkt begrenzt ist. Eine rein zeitliche Begrenzung der Verweildauer im Arbeitsgedächtnis schließt Cowan nicht aus, hierfür lassen sich nach seiner Ansicht jedoch keine eindeutigen Hinweise finden (Cowan, 2005).

In seinem Modell beschreibt er die Menge der aktivierten Daten insofern begrenzt, als diese Aktivierung durch nachfolgende Aktivierungen überschrieben wird, wenn diese der aktuellen Aktivierung in großen Teilen ähnelt. Den Fokus der Aufmerksamkeit wiederum bestimmt Cowan als kapazitiv auf eine feste Menge von Daten begrenzt, die gleichzeitig enthalten sein können. Mögliche Beschränkungen durch fehlende Verarbeitungsstrukturen im Langzeitgedächtnis oder auch durch unterschiedliche Koordinierungsstrategien schließt Cowan nicht aus, stellt sie aber bei seinen Untersuchungen in den Hintergrund.

Eine Beschränkung der aktivierten Teilmenge des Gedächtnisses ist laut Cowan schon vor allem deswegen obligatorisch, um sie von den anderen Inhalten des Gedächtnisses abgrenzen zu können.

Ursprünglich ging Cowan (1999) von einem zeitlich bedingten Abklingen der Aktivierung aus, aktuell sprechen die Versuchsergebnisse aber für eine Interferenz zwischen ähnlichen Daten (Cowan, 2005). Die Frage nach einer möglichen zusätzlichen zeitlichen Beschränkung der aktivierten Daten lässt er offen (Cowan, 2001; Cowan, 2005).

Eine kapazitive Begrenzung der Aufmerksamkeit auf möglicherweise nur ein aktuelles Schema beschreibt Cowan als weit weniger kontrovers. Dieses Schema stellt ein zusammenhängendes Konstrukt von einer begrenzten Anzahl an Informationen und Eigenschaften dar. Informationen, die in keinem Zusammenhang zu diesem Schema stehen, können sich laut Cowan (1999) nicht gleichzeitig im Fokus der Aufmerksamkeit befinden. In einem Versuch wurden Testpersonen auf dem einen Ohr Prosa durch eine männliche Stimme vorgetragen und auf dem anderen Ohr durch eine weibliche Stimme. Hierbei war es für diese Versuchsperson nicht möglich, beiden Aussagen inhaltlich zu folgen (Broadbent, 1958). Unterschiedliche Versuche von Cowan (2001) deuten darauf hin, dass die aktuelle Struktur im Fokus der Aufmerksamkeit wahrscheinlich 4 ± 1 Blöcke zusammengehöriger Daten enthalten kann.

Als Gegenbeispiel wird allerdings aufgeführt, dass es durch Training möglich ist, sich die Fähigkeit anzueignen, zwei komplexe Aufgaben gleichzeitig durchzuführen. Hierbei vermutet Cowan aber eher die Einübung schneller Wechsel der Aufmerksamkeit und gelernte Automatismen.

Als einzige mögliche zeitliche Limitierung der Informationen im Fokus der Aufmerksamkeit erwähnt Cowan Forschungsergebnisse aus dem Bereich der Wachsamkeit. Demzufolge ist es unmöglich, einem Objekt für endlos lange Zeit maximale Aufmerksamkeit zu schenken, da der Wachsamkeitszustand nach ca. einer halben Stunde abnimmt (Davies & Parasuraman, 1982).

3.2.4 Die Rolle des Langzeitgedächtnisses

Cowan vermutet bei der Verknüpfung der Langzeitgedächtnisinhalte eine hierarchische Struktur aus Blöcken von zusammengehörigen Informationen (Cowan, 2005). Beim Erinnern einer Zeile eines Gedichtes würde diese dem Prinzip nach, eine Teilstruktur aus einzelnen Worten darstellen, die bis zu einer gewissen Komplexität im Gedächtnis behalten werden kann. Wenn diese Zeile niedergeschrieben werden soll, zoomt die Aufmerksamkeit auf einzelne Worte, die wiederum Blöcke von Buchstaben darstellen, um so die korrekte Schreibweise zu ermitteln. Für das nächste Wort muss dann wieder zurück auf die gesamte Zeile gezoomt werden, um hier entsprechend

fortzufahren. Beim Heranzoomen verschwinden dann die Inhalte der anderen Blöcke aus dem Fokus der Aufmerksamkeit, da sich diese auf detailliertere Eigenschaften eines Blocks konzentriert.

Informationen aus dem Langzeitgedächtnis können, wenn ein deutlicher Zusammenhang mit der aktuellen Situation einer Person besteht, schnell abgerufen werden, und sind somit mit ähnlicher Effektivität greifbar, wie in aktivierter Form. Cowan (1999) bezeichnet dieses Zusammenspiel mit Langzeitgedächtnisinhalten als „virtual short-term memory“. Langzeitgedächtnisinhalte spielen seiner Meinung nach in den simpelsten Gedächtnisaufgaben eine Rolle. So ist die Gedächtnisspanne für bekannte Worte länger als für unsinnige Buchstabenfolgen.

Neue Zusammenhänge bekannter Informationen werden durch die gleichzeitige Aktivierung dieser Informationen repräsentiert. Gegebenenfalls wird danach der neue Zusammenhang im Langzeitgedächtnis gespeichert. Für die Speicherung deklarativer Zusammenhänge ist hierbei zusätzlich die Präsenz von Aufmerksamkeit notwendig (Cowan, 1999).

3.2.5 Bewertung des Modells

Auch wenn Cowan (1999) mit diesem Modell hervorruft, dass es in manchen Abschnitten vage wirkt, wollte er ein erschöpfendes Modell des Arbeitsgedächtnisses formulieren, in dem die grundlegenden Prozesse enthalten sind, die für alle Arten von Informationen gelten.

Somit unterscheidet sich seine Herangehensweise in der Erklärung der Arbeitsgedächtnisabläufe von der von Baddeley und Logie (vgl. Kapitel 3.1).

Diese legen den Schwerpunkt auf die Begründung von Abweichungen bei Verarbeitungsprozessen, die sich auf verschiedene Arten von Daten beziehen. Cowan hingegen legt den Schwerpunkt auf die Suche nach und Erklärung von Gemeinsamkeiten bei allen Mechanismen, um allgemeingültige Regeln zu finden.

Meiner Ansicht nach lassen sich mit dem „Embedded-Process Model“ einfacher Mechanismen entdecken, die in der Biologie nachgewiesen werden konnten.

So können die Vorgänge der Habituation und Dishabituation zellbiologisch beobachtet werden. Weiterhin kann man auch im menschlichen Gehirn einen aktivierten Teil des Gedächtnisses wiederfinden, in dem man die Menge der Nervenzellen betrachtet, an denen gerade ein Aktionspotential vorliegt.

Deshalb stellt das Modell keine Abstraktion in dem Maße dar, wie das „Multiple-Component Model“. Somit ist das Modell nach meiner Auffassung sowohl leichter

nachvollziehbar als auch besser verifizierbar, da konkrete Abläufe direkt mit Vorgängen in der Biologie verglichen werden können.

Eine Umsetzung in ein computationales Modell wird dadurch erleichtert, dass Cowan konkretere Angaben zu den generellen Mechanismen des Arbeitsgedächtnisses gibt als Baddeley und Logie.

Die Funktionsweise der zentralen Steuereinheit wird von Cowan, ebenso wie im Modell von Baddeley und Logie, nicht genauer beschrieben und offenbart somit auch in diesem Modell eine große Lücke in der Erklärung von Verarbeitungsprozessen.

Den Prozess, Informationen in möglichst zugänglicher Form im Arbeitsgedächtnis zu halten, beschreibt Cowan dadurch, dass die „central executive“ beim Durchsuchen von Datenmengen einzelne Informationen in den Fokus der Aufmerksamkeit befördert. Dadurch werden sie reaktiviert (Cowan, 1999). Dieser Prozess ähnelt dem des Rehearsals, welches von Baddeley und Logie beschrieben wird (vgl. Kapitel 3.1). Durch diese Ähnlichkeit kann er meiner Meinung nach den „primacy“-Effekt genau so beschreiben wie das Rehearsal. Dadurch, dass Informationen, die sich am Anfang einer Liste befinden, schon länger im aktivierten Teil der Gedächtnisses befinden, werden sie insgesamt öfter in den Fokus der Aufmerksamkeit zurück befördert, so dass sie leichter zugänglich sind als Informationen, die noch nicht so lange aktiviert sind.

Der „list length“-Effekt von Cowans Modell kann dadurch erklärt werden, dass Cowan eine zeitliche Beschränkung für die Menge der aktivierten Informationen annimmt (Cowan, 2005). Wenn die Aktivierung verklungen ist, können die Informationen nicht von anderen Informationen im Langzeitgedächtnis unterschieden werden. Wenn also eine Liste eine bestimmte Länge überschreitet, können Informationen verblassen, bevor sie erneut durch die „central executive“ in den Fokus der Aufmerksamkeit gerückt werden konnten (Cowan, 1999).

Erklärungen für die Mechanismen, die im Zusammenhang mit dem „recency“-Effekt sowie für die beschriebenen Fehler beim „immediate serial recall“, bietet das Modell von Cowan allerdings nicht. Cowan (2001) beschreibt zwar die Tatsache, dass die letzten Elemente in Bezug auf die Dauer seit der Aufnahme deutlicher zeitlich von einander getrennt sind als die vorherigen Elemente, als eine mögliche Ursache des „recency“-Effekts, legt aber für sein Modell keine expliziten Mechanismen fest, die einen solchen Effekt erklären würden. Die zugrunde liegende Theorie für diese mögliche Ursache des „recency“-Effekts besagt, dass die einzelnen Elemente

durch eine deutlichere zeitliche Trennung auch deutlicher im Gedächtnis repräsentiert sind (Cowan, 2001). Diese zeitliche Trennung zweier Elemente wird aber im Verhältnis zu der Zeit wahrgenommen, die seit der Präsentation der Elemente vergangen ist. So sind beispielsweise die Repräsentationen von zwei Elementen, die im Abstand von 0,5 Sekunden vor 20 Sekunden präsentiert wurden, weniger deutlich separiert als die Repräsentationen von zwei Elementen, die im Abstand von 0,5 Sekunden vor nur 5 Sekunden präsentiert wurden.

Angaben zu Ursachen für die unterschiedlichen Fehlerarten beim „immediate serial recall“ werden in Cowans Arbeitsgedächtnismodell nicht gemacht. Trotzdem sind die Lücken in der Modellbeschreibung groß genug, um wie im Modell von Baddeley und Logie (vgl. Kapitel 3.1) Mechanismen nicht auszuschließen, die sowohl den „recency“-Effekt als auch die beiden beschriebenen Fehlerarten erklären könnten.

3.3 „Working Memory in Everyday Skilled Performance“ von K. Anders Ericsson und Peter F. Delaney

Das theoretische Arbeitsgedächtnismodell von Ericsson und Delaney entwickelte sich aus dem Modell des „Skilled Memory“ von Chase und Ericsson (1981).

Anstoß für die Entstehung des ursprünglichen Modells waren Beobachtungen von Testergebnissen bei Gedächtnisaufgaben, die die vermuteten Arbeitsgedächtnisbeschränkungen bei weitem übertrafen. So waren beispielsweise einzelne Versuchspersonen in der Lage, präsentierte Ziffernfolgen von einer Länge von bis zu 80 Ziffern korrekt zu wiederholen, was zu den durchschnittlich üblichen Sequenzlängen von 7 Ziffern eine mehr als 1000 prozentige Steigerung darstellt.

Aus diesen Leistungen und aus den eigenen Aussagen der Testpersonen über die Methoden zur Speicherung folgerten sie, dass Experten offenbar für spezielle Fachgebiete schnell auf ihr Wissen aus dem Langzeitgedächtnis zugreifen und es so für Arbeitsgedächtnisaufgaben nutzen konnten (Ericsson & Delaney, 1999).

Durch weitere Versuche mit allgemeineren Aufgaben, die eher dem täglichen Leben entsprachen, schlossen Ericsson und Delaney, dass das Langzeitgedächtnis generell nicht vollständig von den Prozessen des Arbeitsgedächtnisses getrennt werden kann.

Teile des Langzeitgedächtnisses können demzufolge bei Routineaufgaben als Erweiterung zum gängigen Arbeitsgedächtnis eingesetzt werden (Ericsson & Kintsch, 1995).

Zur Differenzierung und Vergleichbarkeit führten sie die Begrifflichkeiten des „Short Term - Working Memory“ kurz ST-WM für die bisherige Sichtweisen des zeitlich und inhaltlich begrenzten Arbeitsgedächtnisses ein und den des „Long Term - Working Memory“ kurz LT-WM für die zusätzlichen Strukturen, die für die Kontrolle und Verarbeitung von Informationen aus dem Langzeitgedächtnis bei kognitiven Prozessen verantwortlich sind.

Insgesamt definieren sie nun das Arbeitsgedächtnis als die Struktur der Mechanismen zur Kontrolle und Selektion von Informationen, die für die Lösung einer beliebigen Aufgabe benötigt werden (Ericsson & Delaney, 1999).

Eigenschaften anderer Modelle, wie die zeitliche und inhaltliche Beschränktheit, werden von ihrer Definition nicht erfasst.

Zur Erforschung der Mechanismen und Eigenschaften des Arbeitsgedächtnisses unterscheiden Ericsson und Delaney drei Herangehensweisen:

Die Untersuchung grundlegender Kapazitäten, die Untersuchung des Einflusses von Stördaten und die Untersuchung der Leistung von Experten.

3.3.1 „Short Term – Working Memory“

Das „Short Term - Working Memory“ wird von Ericsson und Delaney als konsistent zu den Modellen sowohl von Cowan als auch von Baddeley und Logie beschrieben. Dabei stellt das ST-WM nur einen Teil des Kurzzeitgedächtnisses dar, der für die Informationsverarbeitung zuständig ist. Das ST-WM wird bei vielen Gedächtnisaufgaben mit beansprucht, aber nur in Ausnahmefällen ausschließlich eingesetzt. Die Kapazitäten sind analog zu den Untersuchungen von George Miller (1956) sowohl zeitlich als auch inhaltlich erheblich limitiert (Chase & Ericsson, 1982).

Miller (1956) fand heraus, dass die Kapazität für die Merkfähigkeit von Datenblöcken verschiedenster Art bei allen Versuchspersonen im Bereich 7 ± 2 lag.

Baddeley (1986) begründete dies mit dem Prozess des inneren sprachlichen Wiederholens (Rehearsal) und erklärte individuelle Unterschiede mit unterschiedlicher Sprechgeschwindigkeit.

Da bei kognitiven Vorgängen aber neben den Ressourcen des ST-WM auch auf das LT-WM zurückgegriffen wird, sagen diese Beschränkungen nur wenig über die Leistungsfähigkeit der Testpersonen bei Aufgaben des alltäglichen Lebens aus.

Beim Erlernen der Fertigkeiten, die die Nutzung von LT-WM erlauben und deren Effektivität steigern sind die Personen aber noch auf die Speicher- und Verarbeitungsmechanismen des ST-WM angewiesen. Diese beschreiben Ericsson und Delaney (1999) entsprechend dem Modell von Baddeley und Logie (vgl. Kapitel 3.1). So kommen die erwähnten beschränkten Speichermöglichkeiten bei abstrakten Laborversuchen zustande, die oft mit dem Ziel konzipiert werden, Vorteile durch Wissen und Erfahrungen auszuschließen.

Dadurch bleibt nach den Aussagen von Ericsson und Delaney aber ein wichtiger Teil des Arbeitsgedächtnisses unberücksichtigt.

3.3.2 Hinweise für zusätzliche Kapazitäten

Personen mit niedrigerer Kapazität des „phonological loop“ auf Grund von Hirnschädigungen zeigten in alltäglichen Aufgaben, wie etwa Textverständnis, normale Leistungen. Dies spricht laut Ericsson und Delaney für zusätzliche Speichermöglich-

keiten des Arbeitsgedächtnisses außerhalb des „phonological loop“ (Ericsson & Delaney, 1999). Weiterhin zeigten wiederholte Versuche mit den gleichen Testpersonen, dass sie in der Lage waren durch Training ihre Kapazität um bis zu 1000 % für die aktuelle Aufgabe zu erhöhen. Diese zusätzliche Kapazität basiert aber nicht auf einer Erweiterung des „phonological loop“, da dieser bei untrainierten Aufgaben wieder die ursprünglichen Leistungsmerkmale zeigte.

Sowohl die normalen Leistungen der Personen mit geschädigtem „phonological loop“ als auch die Möglichkeit die Leistungsfähigkeit durch Training zu erhöhen, widersprechen damit laut Ericsson und Delaney (1999) der Annahme, dass die begrenzte Kapazität des Kurzzeitgedächtnisses auch die Leistungen in allen kognitiven Aufgaben limitiert.

Als weiteres Indiz wird schon von Miller (1956) erwähnt, dass Versuchspersonen, die in der Lage sind, sich durchschnittlich sechs oder sieben zusammenhanglose Buchstaben zu merken, sich eine viel größere Anzahl einprägen können, wenn die Buchstaben Wörter bilden. Dann nämlich steigt die Menge auf sechs bis sieben Wörter, was wiederum 20 bis 40 verschiedenen Buchstaben entspricht. Das Wissen, dass es sich bei den Buchstaben um Wörter handelt, stammt wiederum aus dem Langzeitgedächtnis (Ericsson & Delaney, 1999).

Bei der Erforschung grundlegender Kapazitäten des Arbeitsgedächtnisses zeigten sich nach Ericsson und Delaney, dass die von Miller (1956) genannte Kapazität von 7 ± 2 Datenblöcken im Arbeitsgedächtnis bei bestimmten Aufgaben um ein vielfaches überschritten werden kann. Dies erfolgt ihrer Meinung nach durch das Einbeziehen von Langzeitwissen. Die feste Beschränkung bezieht sich daher nur auf das ST-WM.

Bei Untersuchungen mit Stördaten wird versucht, das Arbeitsgedächtnis vor einem Test mit Daten zu füllen bzw. mit einer zweiten Aufgabe zu beschäftigen, um so den Einfluss von einzelnen Komponenten des Arbeitsgedächtnisses auf den eigentlichen Test herabzusetzen. Beispielsweise könnte die sekundäre Aufgabe, sich eine Ziffernfolge zu merken, den „phonological loop“ beschäftigen und ihn somit als Einflussgröße für die Leistungen in der Primäraufgabe minimieren.

Hier zeigen sich zwar schlechtere Speicherfähigkeiten bei Laborversuchen, aber nicht in dem Maße, wie die geschätzte Kapazität des Arbeitsgedächtnisses es erwarten ließe. Noch deutlicher wird dies bei alltäglicheren Tests wie Textverständnis, wo der Einfluss noch weiter abnimmt. Experten zeigten sogar in ihren Bereichen keine merkliche Interferenz (Ericsson & Delaney, 1999).

Als weiteres Indiz dafür, dass die zeitlich begrenzte Speicherung bei gelernten Fähigkeiten wie Lesen eine geringe Rolle spielt, führen Ericsson und Delaney (1999) einen Versuch auf, bei dem die aktuelle Aufgabe plötzlich mit einer ablenkenden Aufgabe unterbrochen wird und anschließend von der Testperson zu Ende durchgeführt werden soll. Ein zeitlich und inhaltlich beschränktes Arbeitsgedächtnis würde sich mit den Daten der neuen Aufgabe füllen und das Abschließen der ursprünglichen Tätigkeit unmöglich machen. Doch stattdessen konnten die Testpersonen bei kurzen Unterbrechungen nach wenigen Sekunden ihre Aufgabe an der Stelle fortführen, an der die Unterbrechung stattfand.

Ericsson und Delaney (1999) argumentieren, dass dies nur durch Kodierung der Informationen im Langzeitgedächtnis möglich ist.

3.3.3 „Long Term – Working Memory“

Im Gegensatz zu den sonst üblichen Arbeitsgedächtnistests versuchten Ericsson und Delaney bei ihren Untersuchungen nicht, aus einer breiten Masse an Ergebnissen Rückschlüsse auf allgemein gültige Angaben zu den Beschränkungen des Arbeitsgedächtnisses zu bilden, sondern sie konzentrierten ihr Augenmerk auf genau die Versuchspersonen, deren Leistungen sich deutlich von der Masse abhoben.

Die kognitiven Fähigkeiten von Experten bezeichnen Ericsson und Delaney (1999) deswegen als für Untersuchungen interessant, da sie in reproduzierbarer Form Leistungsobergrenzen widerspiegeln.

Es ist laut Ericsson und Delaney möglich, diese Fähigkeiten durch Laborversuche aufzuzeigen und unter kontrollierten Bedingungen den Fähigkeiten anderer Testpersonen gegenüber zu stellen.

Sie analysierten dabei eher alltägliche Aufgaben, anstatt speziell entwickelte Laborversuche, die versuchen, Einflüsse des Langzeitgedächtnisses auszuschließen.

Um zu vergleichen, auf welche Weise eine Aufgabe gelöst wurde, verlassen sich Ericsson und Delaney auf die anschließenden Aussagen der Testpersonen.

Die Überlegenheit von Experten äußert sich bei diesen Tests durch spezielle Fähigkeiten zu planen, zu schlussfolgern und vorherzusehen (Ericsson & Delaney, 1999).

Basierend auf diesen erwähnten Testergebnisse schlugen Ericsson und Kintsch (1995) den Begriff des „Long Term - Working Memory“ vor.

Die Nutzung dieses Teils des Arbeitsgedächtnisses basiert auf so genannten „*memory skills*“, die vorher erlernt werden müssen.

Diese Fähigkeiten erlauben Experten beispielsweise, in einer bestimmten Domäne Informationen so schnell im Langzeitgedächtnis zu speichern und auch abzurufen, dass es während einer Denkaufgabe effektiv genutzt werden kann. Dies geschieht durch eine Struktur von Abrufhinweisen, die jeweils zur Wiederherstellung bestimmter Informationen dienen. Die Datenmenge, auf die Arbeitsgedächtnisprozesse innerhalb kurzer Zeit zurückgreifen können, unterliegt auf diese Weise keiner so starken Begrenzung, wie es vorherige Modelle beschreiben.

Interferenzen von ähnlichen Daten im Langzeitgedächtnis durch gleiche Abrufhilfen kann das LT-WM durch die Nutzung von verschiedenen redundanten Abrufhilfen, die als Zusatzinformationen gespeichert werden, entgegenwirken.

Außerdem müssen Experten gespeicherte Informationen unterschiedlicher Domänen auf verschiedenartige Weise verarbeiten und analysieren. Somit bilden sich auch hier unterschiedliche Strategien und Mechanismen zur Speicherung und Nutzung je nach Anforderungen der Domäne (Ericsson & Delaney, 1999). So erklären Ericsson und Delaney verschiedene Fähigkeiten von Experten.

Ericsson und Delaney betonen bei ihrem Modell, dass es sich bei der überdurchschnittlichen Leistungsfähigkeit des Arbeitsgedächtnisses für spezielle Bereiche ausschließlich um erlernte Fähigkeiten handelt. So könne jeder nach dem Prinzip von Gedächtnishilfen und Eselsbrücken die Menge an Daten erhöhen, die er sich kurzzeitig merken kann. Aus den Untersuchungen von so trainierten Testpersonen und Experten ging hervor, dass die Methoden und Abläufe beim Speichern und Erinnern der beiden Personengruppen vergleichbar sind. Wenn Informationen mit Langzeitwissen verknüpft und durch ihre Eigenschaften miteinander verbunden werden, sagt die Kapazität des ST-WM nichts mehr über die Leistungsfähigkeit einer Person bei alltäglichen Aufgaben aus. Die Ursache für diesen fehlenden Zusammenhang ist, dass eine Speicherung im ST-WM beispielsweise im „phonological loop“ keinen semantischen Bezug zur Aufgabe herstellt und deswegen nur für eine kurzzeitige Speicherung nützlich ist, nicht aber für Aufgaben, die ein Verstehen oder Interpretieren der Daten voraussetzen. Auf diese Weise argumentieren Ericsson und Delaney (1999) beispielsweise, dass der „reading span“-Test kein Indiz für die ST-WM Kapazität ist, da die zu merkenden Wörter mit Bedeutungen aus dem Langzeitgedächtnis verknüpft werden. Bei diesem Test werden die Versuchspersonen aufgefordert, mehrere Sätze laut vorzulesen und sich dabei jeweils das letzte Wort eines jeden Satzes zu merken.

Dieses stützen sie durch die Berichte der Versuchspersonen (Daneman & Carpenter, 1980). Unterschiedliche Ergebnisse bei diesen Tests zeigen dementsprechend nur unterschiedliche Fähigkeiten, Informationen effizient im LTM zu speichern.

Laut Ericsson und Delaney basiert der „reading span“-Test auf denselben Mechanismen wie das Textverständnis beim Lesen. Unterschiedliche Leistungen basieren hier zum einen auf dem thematischen Hintergrundwissen der Testperson sowie seiner Sprachfertigkeiten wie z.B. Wortschatz und Satzbauwissen. Aus dieser ihrer Meinung nach falschen Interpretation des Tests weisen sie auf das Problem vieler Laborversuche hin, dass nur Ergebnisse betrachtet werden und nicht der Lösungsweg und die Lösungsfindung.

Bei der Aufgabe, sich eine Ziffernfolge zu merken, konnten Chase und Ericsson (1981) feststellen, dass mit zunehmendem Training die Länge der korrekt wiederholten Ziffernfolge immer mehr zunahm. Einzelne Personen waren in der Lage, sich bis zu 80 Ziffern zu merken. Diese gesteigerten Fähigkeiten wirkten sich aber nicht auf die Ergebnisse anderer Tests aus, bei denen andere kognitive Prozesse beansprucht wurden oder auch andere Arten von Daten behandelt wurden. Dafür konnten aber viele der Listen auch noch längere Zeit nach den Tests von den geübten Testpersonen wiederholt werden. Aus diesen Gründen argumentieren Ericsson und Delaney (1999) erneut für eine Speicherung im Langzeitgedächtnis.

Dabei werden nach ihren Angaben Teile der Ziffernfolge jeweils mit bestimmten Erfahrungen oder Informationen im Langzeitgedächtnis verbunden. Hierfür muss jedoch nicht die explizite Ziffernfolge als Vorwissen im Gedächtnis vorhanden sein sondern nur mögliche Bedeutungen, die die Zahl haben könnte. Weiterhin werden die Teilsequenzen auf Eigenschaften und Beziehungen untereinander untersucht. Diese werden dann in Form von Eselsbrücken als zusätzliche Abrufhinweise verknüpft (Ericsson & Delaney, 1999).

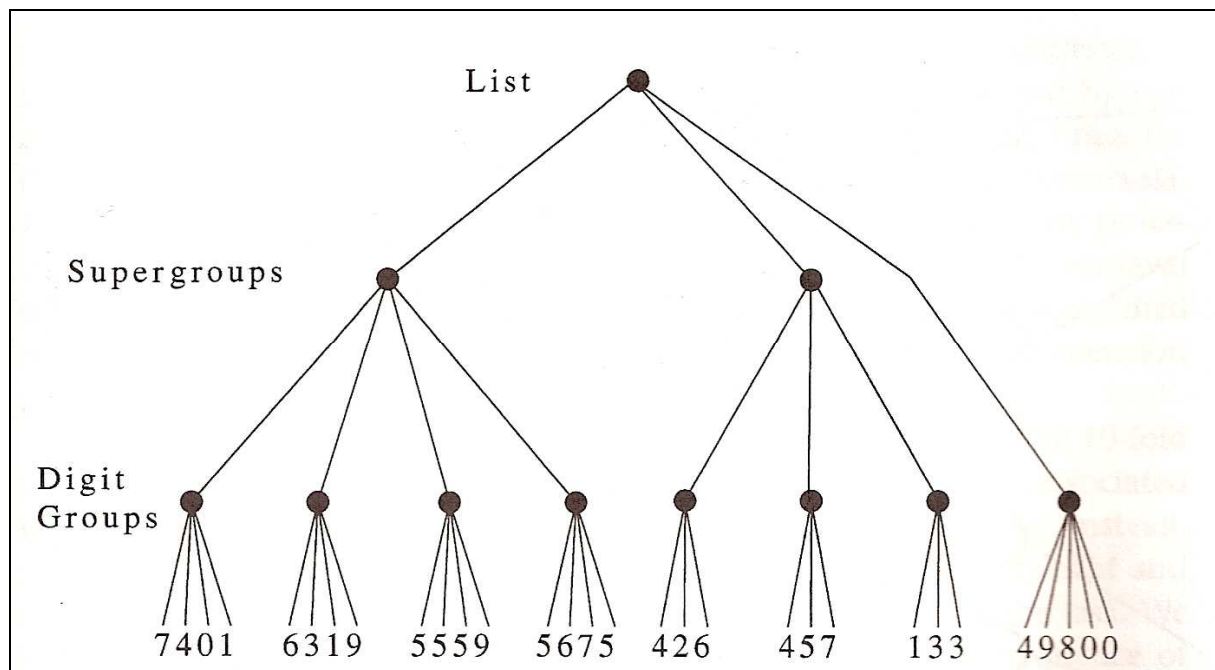


Abbildung 6: Struktur der Abrufhinweise im LT-WM für eine Ziffernfolge von 30 Ziffern (Ericsson & Delaney, 1999)

Wenn Zahlen präsentiert werden, mit denen keine passenden Erfahrungen verknüpft werden können, fallen die Ergebnisse deutlich schlechter aus. Mit zunehmendem Training finden die Testpersonen aber neue Zuordnungsmethoden, so dass auch diese Zahlen mit einer Bedeutung aus dem Langzeitgedächtnis versehen werden.

Aus diesen Bedeutungen und Abrufhinweisen wird dann in Gedanken eine Struktur gebildet, die die gesamte Ziffernfolge repräsentiert. Diese Struktur kann mit Hilfe der Aussagen der Testpersonen bildlich wie in Abbildung 6 zusammengefasst werden. Knoten stellen eine Zusammenfassung der verknüpften Daten zu einer Gruppe mit bestimmter Bedeutung dar. Ericsson und Delaney haben Testpersonen einige Zeit nach dem Test die Abbildungen vorgelegt, ohne dass die expliziten Ziffern an den Enden der Äste dargestellt waren. Trotzdem waren die Testpersonen in der Lage, lediglich anhand der Struktur sowohl die Bedeutungen der Knoten als auch die entsprechenden Ziffernfolgen wiederzugeben.

Da diese Struktur von den Testpersonen ohne Überlegen interpretiert werden konnte, argumentieren Ericsson und Delaney (1999), dass die tatsächliche Verknüpfung der Informationen im Gedächtnis der gezeichneten Struktur entspricht.

Dass das LT-WM nicht nur zur Speicherung sondern auch zum Planen und Schlussfolgern genutzt werden kann, zeigen Ericsson anhand der Fähigkeiten von

Schachexperten. Diese können ihre innere Repräsentation einer Aufstellung nutzen, um vorteilhafte mögliche Züge zu überlegen. Dabei würde eine unflexible Repräsentation einer Aufstellung als einzelner Block die beobachtete Möglichkeit eines Überdenkens und gegebenenfalls Umentscheidens zu einer besseren Lösung nicht ermöglichen (de Groot, 1946/1978). Saariluoma (1989) zeigte, dass die überragende Fähigkeit von Schachexperten sich eine Schachaufstellung zu merken auf dem Verbinden einzelner Positionen mit Bedeutungen und Beziehungen beruht. So konnten Schachexperten eine gemerkte Aufstellung von einer nicht gezeigten Aufstellung leichter unterscheiden, wenn der Unterschied spielbedingt größere Bedeutung hat. Dadurch erzielten sie bei zufälligen Aufstellungen schlechtere Ergebnisse als bei tatsächlich möglichen Spielaufstellungen.

3.3.4 Bewertung des Modells

Der Grundgedanke des Modells von Ericsson und Delaney ist unter anderem, dass sich das Arbeitsgedächtnis nicht auf das ST-WM beschränken lässt und dass die bisher gängigen Beschränkungen für ihr Modell vom Arbeitsgedächtnis nicht mehr gültig sind.

Obwohl LT-WM ursprünglich entwickelt wurde, um die Performance von Experten zu erklären, lässt es sich auf viele alltägliche Aufgaben anwenden.

Das theoretische Modell wird, wie bei Cowans „Embedded-Process Model“, so allgemein formuliert, dass es sich auf alle Mechanismen kognitiver Verarbeitung ausgewählter Daten verwenden lässt (Ericsson & Delaney, 1999). Auch Ericsson und Delaney versuchen generelle Mechanismen aufzuzeigen, die für alle Arten von Informationen in gleicher Weise gelten.

Als Ursache für das Vergessen von Informationen im Kurzzeitgedächtnis geben Ericsson und Delaney (1999) Interferenzen zwischen den gespeicherten Daten an. Dadurch würden neue Informationen ältere Informationen überschreiben, wenn sie sich mit diesen überschneiden. Auf diese Weise kann das Modell erklären, wie der „recency“-Effekt beim „immediate serial recall“ zustande kommt. Elemente, die später in der Liste auftauchen, überschreiben teilweise die Elemente, die sich an vorangehenden Positionen befinden. Somit sind die Repräsentationen der letzten Elemente vollständiger erhalten als die der ersten Elemente. Das Modell schließt zusätzliche Mechanismen, die den „recency“-Effekt verursachen können, nicht aus (Ericsson, Kintsch, 1988).

Zur Erklärung des „primacy“-Effekts werden im Modell keine Eigenschaften oder Mechanismen genannt.

Den „list length“-Effekt erklärt das Modell dadurch, dass es die Beschränkungen für das Kurzzeitgedächtnis entsprechend der Modelle von Baddeley und Logie (vgl. Kapitel 3.1) und Cowan (vgl. Kapitel 3.2) übernimmt. Zusätzlich interpretiert es individuelle Unterschiede in den Fähigkeiten, sich Listen von einer bestimmten Länge merken zu können, als angelegte „memory skills“. So können sich nach den Aussagen von Ericsson und Kintsch (1999) Experten längere Listen merken, wenn sie in ausreichendem Training Speicherstrategien entwickelt haben (Ericsson, Kintsch, 1999).

Da sich das Modell darauf konzentriert, den Einfluss des Langzeitgedächtnisses bei Arbeitsgedächtnisprozessen zu beschreiben und die Mechanismen des Kurzzeitgedächtnisses dabei weitestgehend außer Acht lässt, werden auch keine Einzelheiten beschrieben, die zu den beschriebenen Fehlerarten führen.

Ich halte das Modell aber insofern für besonders erwähnenswert, da es meiner Ansicht nach eine plausible Struktur für die Einbeziehung des Langzeitgedächtnisses in Arbeitsgedächtnisprozesse bietet. Im Modell von Baddeley und Logie (vgl. Kapitel 3.1) werden Langzeitgedächtnisinhalte über die Zwischenstufe des „episodic buffer“ verarbeitet und im Modell von Cowan (vgl. Kapitel 3.2) in Form von „virtual short-term memory“. Diese Verfahren stellen meiner Meinung nach eine sehr starke Trennung zwischen Lang- und Kurzzeitgedächtnis dar, die ich in solcher Form im biologischen Vorbild nicht auffinden kann.

3.4 „A Recurrent Neural Network Model“ von Matthew M. Botvinick und David C. Plaut

Im Gegensatz zu den vorangehenden Modellen handelt es sich bei dem Ansatz von Botvinick und Plaut (2006) um ein komputationales Modell. Sie entwarfen mit ihrer Arbeit ein neues Modell zur Erklärung der Mechanismen beim „serial recall“. Bei diesem Versuch werden Abfolgen von Objekten gezeigt, die die Testpersonen im Anschluss wiederholen sollen.

„Serial recall“-Aufgaben unterscheiden sich in den Objekten, die wiederholt werden sollen und in den Vorgaben für die Beantwortungsphase (vgl. Kapitel 2.2). Es kann sich bei den Objekten zum Beispiel um gezeigte Bilder, gesprochene Worte oder einzelne Abbildungen von Buchstaben oder Zahlen handeln. Die Beantwortung kann unmittelbar im Anschluss an die Präsentation folgen oder auch nach einer vorher festgelegten Zeit. Im Gegensatz zum so genannten „free recall“ müssen die Objekte in der Reihenfolge wiederholt werden, in der sie präsentiert wurden (Baddeley, 1999).

Hierbei berufen sich Botvinick und Plaut (2006) auf Ergebnisse der Untersuchungen der Funktionsweise neuronaler Netze in den Bereichen der Psychologie und der Neurowissenschaften.

Obwohl viele Wissenschaftler davon ausgingen, dass man durch ein neuronales Netzwerk, welches auf Aktivierung basiert, niemals die empirischen Effekte beim „immediate serial recall“ abbilden könne, stellen Botvinick und Plaut (2006) in ihrer Arbeit eine derartige Struktur vor. Sie zeigen, dass solch ein neuronales Netz durchaus eine breite Palette empirischer Daten simulieren kann.

Per Computersimulation wenden sie ein *rekurrentes neuronales Netz* auf die Aufgabe des „immediate serial recall“ an. Rekurrente Verbindungen bewirken, dass die Zustände des Netzes nicht nur aus gegebenen Inputs berechnet werden, sondern per Rückkopplungen von vorherigen Zuständen des Netzes mit beeinflusst werden.

Eine der ersten Theorien zu den Mechanismen, die dem „serial recall“ zu Grunde liegen, stammt von Donald Hebb (1949). Dieser vermutete dahinter so genannte „activity traces“, die eine Sequenz als bleibende Spuren der Aktivierung der einzelnen Elemente darstellen. Zur Aufrechterhaltung dieser Spuren wurden schon in frühen Studien rekurrente Verknüpfungen vorgeschlagen (Conrad, 1959; Lashley, 1951).

Heutzutage gilt es laut Botvinick und Plaut (2006) als allgemein akzeptiert, dass die aufrechterhaltenen Aktivierungen bei den Vorgängen des Arbeitsgedächtnisses eine zentrale Rolle spielen und dass diese durch rekurrente synaptische Verbindungen unterstützt werden.

Botvinick und Plaut unterscheiden zwei Herangehensweisen im Bezug auf das Verständnis für serial recall im Kurzzeitgedächtnis. Zum einen den gewichts- oder kontextbasierten Ansatz und zum anderen den aktivierungsbasierten Ansatz. In ihrer Arbeit argumentieren sie für den aktivierungsbasierten Ansatz eines rekurrenten neuronalen Netzes.

Bei der Implementierung haben Botvinick und Plaut versucht, überall Vereinfachungen zu finden, um die theoretischen Annahmen auf einen Kern zu begrenzen.

3.4.1 Aufbau des Modells

Der Aufbau des Modells, das Botvinick und Plaut (2006) in ihrem Artikel vorstellen entspricht dem eines künstlichen neuronalen Netzes. Einfache Verarbeitungselemente mit sich verändernden Aktivierungen sind über gewichtete Verbindungen miteinander verknüpft. Die Elemente sind in drei Gruppen unterteilt. Die erste Gruppe bildet eine Input-Schicht, an der Eingaben des Netzes angelegt werden können, und die zweite Gruppe eine Output-Schicht, an der die Ausgabe des Netzes abgelesen werden kann. Die dritte Schicht ist eine verdeckte Verarbeitungsschicht, die ihre Aktivierung aus dem Zustand der Input-Schicht und den entsprechenden Verbindungsgewichten berechnet und auf diese Weise an die Output-Schicht weitergibt.

In der Input-Schicht existiert ein Element, das mit seiner Aktivierung die Wiederholungsphase anzeigt und in der Output-Schicht ein Element, das das Ende der Liste repräsentiert.

Rekurrente Verbindungen existieren in diesem Netzwerk zwischen den einzelnen Elementen der verdeckten Schicht sowie von der Output-Schicht zurück zur verdeckten Schicht. Auf diese Weise beeinflusst der vorangehende Zustand des Netzes den nächsten Zustand und eine Ausgabe nimmt Einfluss auf die folgenden Zustände. Diese rücklaufenden Verbindungen werden mit einer Verzögerung von einem Zeitschritt verarbeitet. Das bedeutet, dass aus dem ersten Input der Zustand des ganzen Netzes berechnet wird, ohne dass die rekurrenten Verbindungen einen Einfluss haben. Beim folgenden Input wird bei der Berechnung des neuen Zustandes jedoch der vorherige über die rekurrenten Verbindungen hinzugerechnet.

Verbindungen bestehen immer zwischen allen Elementen einer Schicht und allen Elementen der verbundenen Schicht.

Die Verbindungsgewichte wurden in einer Trainingsphase vor der eigentlichen Berechnungsaufgabe mit Trainingsdaten optimiert. Die Trainingsdaten bestanden aus Sequenzen unterschiedlicher Länge und die berechneten Ausgaben wurden genutzt um die Gewichte anzupassen. Hierbei wurde ein *Backpropagation-Algorithmus* für rekurrente Netze verwendet (Williams & Zipser, 1995). Das Training wurde beendet, wenn die aktuelle Konfiguration der Gewichte eine festgelegte Erfolgsquote beim Erinnern von Sequenzen aufzeigte. Die Wahrscheinlichkeit, dass in der eigentlichen Testphase die gleichen Listen auftauchten, die auch in der Trainingsphase vorkamen war laut Botvinick und Plaut (2006) sehr gering.

Während der eigentlichen Testphase wurden dann die Gewichte nicht weiter verändert. So kann gezeigt werden, dass die Leistungen des Modells beim „immediate serial recall“ nur auf Aktivierungen basieren, nicht aber auf sich verändernden Gewichten.

Um die Trainingsphase des Modells biologisch plausibel zu machen, begründen Botvinick und Plaut sie mit der Aneignung des Menschen von Aufgaben wie Sprachverstehen, die auf dem Reproduzieren von sequenziellen Daten basieren. Diese werden mit zunehmender Erfahrung trainiert.

3.4.2 Funktionsweise des neuronalen Netzes

Eine Eingabe stellte bei dem Modell ein einziges aktiviertes Element der Input-Schicht dar, während alle anderen Elemente eine Aktivierung von Null besitzen. So sollte verhindert werden, dass ungewollte Ähnlichkeiten der Eingaben das Ergebnis verfälschten. Weiterhin diente diese Kodierung der Simplifizierung und somit Interpretierbarkeit des Modells. Laut Botvinick und Plaut hätten aber auch verteilte Repräsentationen an der Input-Schicht zu den gleichen Ergebnissen geführt. Bei einer separaten Versuchsreihe zur Modellierung der Einflüsse von Ähnlichkeiten wurden zweidimensionale Repräsentationen genutzt. Dabei unterschieden sich alle Eingaben in der ersten Dimension, konnten sich aber in der zweiten Dimension gleichen. Dadurch wurden Überschneidungen von 0 % und 50 % möglich.

Die Aufgabe für das Netz war in allen Versuchsreihen „immediate serial recall“. Hierbei wurden in mehreren Durchläufen verschiedene Einflüsse auf die Performance simuliert.

Die Antwort des Netzes stellte eine Reihenfolge von Ausgaben der Output-Schicht dar, die in ihrer Länge der Länge der ursprünglichen Liste glich. Eine vorherige Aktivierung des Elementes, welches das Ende der gemerkten Liste darstellt, wurde ignoriert. Botvinick und Plaut verglichen diese Vorgehensweise mit einer Merkaufgabe, bei der auf einem Antwortzettel die feste Anzahl der Listenelemente vorgegeben ist.

Zusätzliche Simulationen, bei denen kürzere Ausgaben erlaubt waren zeigten, dass das Gesamtmuster der Versuchsergebnisse sich nicht unterschied.

Als Ausgabe eines Zeitschrittes wurde das am stärksten aktivierte Element der Output-Schicht ausgewählt.

Bei der Auswahl der Phänomene, welche Botvinick und Plaut mit ihrem Modell aufzeigen wollten, haben sie nach drei Kriterien entschieden. Zum Ersten wählten sie grundlegende Effekte, die als Bewertungsmaßstab für Modellimplementierungen gelten und zum Zweiten Effekte, die als Argumente gegen die Nutzungsmöglichkeit rekurrenter Netze für „serial recall“ galten. Drittens wählten sie Phänomene, die einzigartige Eigenschaften des aktuellen Modells aufzeigen, welche Botvinick und Plaut (2006) in ihrer Integration von Hintergrundwissen bei der Wiederholung von Listen sehen.

Um Fehler des Modells zu produzieren wurden zufällige Werte als eine Art Hintergrundrauschen auf die Aktivierung der verdeckten Schicht addiert bzw. subtrahiert.

Der beschriebene Aufbau führt dazu, dass das Netzwerk in der Lage ist, in seiner verdeckten Schicht die Repräsentation eines Listenelementes zu erzeugen. Diese Repräsentation beinhaltet neben der Information, um welches Element es sich handelt, zusätzlich die Position des Elementes innerhalb der Liste. Die gesamte Liste, die bisher gelernt wurde, wird als Überlagerung aller Repräsentationen der einzelnen Elemente dargestellt. Dies wird im Modell durch die Summation der einzelnen Muster umgesetzt. Diese Summation kann aber laut Botvinick und Plaut (2006) ohne weiteres wieder zurück in die Repräsentationen von einzelnen Listenelementen zerlegt werden.

Bei der Analyse dieser einzelnen Repräsentationen stellten Botvinick und Plaut fest, dass Element und Position verbunden kodiert sind, und dass sie unabhängig von den anderen Elementen der Liste kodiert werden. Daraus folgt, dass ein Element an einer bestimmten Position eine andere Repräsentation darstellt, als das selbe Element an einer anderen Position. Es spielt aber für die Repräsentation keine Rolle,

welche Elemente ansonsten in der Liste enthalten sind. Repräsentationen, die das gleiche Element an einer anderen Listenposition darstellen, weisen aber gewisse Ähnlichkeiten auf, wie auch Repräsentationen unterschiedlicher Elemente an derselben Listenposition. Somit entstehen Ähnlichkeiten der repräsentierten Elemente einerseits durch Ähnlichkeit der Elemente an sich und andererseits durch benachbarte Listenpositionen.

3.4.3 Ergebnisse der Simulationen

Das Modell zeigt bei unterschiedlichen Listenlängen einen sigmoidalen Abfall der Anzahl der korrekt wiederholten Listen bei steigender Länge auf. Dies entspricht den Untersuchungsergebnissen von Crannell und Parrish (1957). Botvinick und Plaut erklären diesen Effekt damit, dass die Repräsentationen immer eine Summation aller Listenelemente darstellt. Bei zunehmenden Listenlängen entstehen also zunehmend unschärfere Repräsentationen einzelner Elemente.

Das jeweils erste und letzte Listenelement wurde bei den Testdurchläufen im Durchschnitt häufiger korrekt wiederholt, als die anderen Listenelemente. Diese so genannten primacy- und recency-Effekte resultieren laut Botvinick und Plaut in ihrem Modell daraus, dass Fehler oftmals aus der Verwechslung von Elementen mit ihren Nachbarn in der Liste resultieren. Das erste sowie das letzte Element haben den Vorteil, nur einen Nachbarn in der Liste zu besitzen. Weiterhin haben Elemente die sich am Anfang einer Liste befinden den Vorteil, dass sie kodiert werden, wenn sich noch keine bzw. nur wenige Repräsentationen von vorangehenden Elementen in der verdeckten Schicht befinden. Zu den besseren Ergebnissen beim Erinnern der letzten Listenelemente trägt bei, dass sich bei der aktuellen Summation der Repräsentationen die bereits wiederholten Elemente von den noch folgenden unterscheiden lassen. Bei den letzten Elementen bedeutet das, dass nur wenige bzw. keine Elemente vorhanden sind, mit denen sie leicht verwechselt werden können (Botvinick & Plaut, 2006).

Durch die Verwendung von sich überschneidenden Repräsentationen der Netzeingaben konnten Botvinick und Plaut zeigen, dass bei Sequenzen, deren Elemente sich in der Repräsentation alle zu 50 % überschneiden, eine deutlich höhere Fehlerquote resultierte als bei Sequenzen, in denen keine Überschneidungen existierten. Bei alternierender Eingabe von Elementen, die sich überschneiden, und Elementen, die sich nicht überschneiden, zeigte sich ebenfalls ein alternierender Verlauf der Fehlerzahlen. Dabei wurde deutlich, dass im gezeigten Modell keine feste Verkettung

zwischen den einzelnen Elementen stattfindet, da dieses dazu führen würde, dass der Fehler bei Auftreten eines unverwechselbaren Elementes trotzdem hoch bleibt. Als Hintergrund für diese Reaktionen des Modells erklären Botvinick und Plaut, dass sich überschneidende Elemente ähnliche interne Repräsentationen besitzen, was zu leichteren Verwechslungen führt. Da die einzelnen Elemente aber unabhängig kodiert werden, resultiert aus einer Verwechslung eines Elementes nicht auch ein falsches folgendes Element. Dadurch kommt die alternierende Fehlerkurve bei alternierenden verwechselbaren und unverwechselbaren Eingaben zustande.

Als letztes zeigten Botvinick und Plaut (2006) die Wirkung von Langzeitwissen in Form einer künstlichen Grammatik auf. Für das Training des Modells bedeutete dies, dass manche Reihenfolgen als wahrscheinlicher galten und somit auch häufiger in der Trainingsphase vorkamen. Unwahrscheinlichere Reihenfolgen wurden seltener trainiert. Dadurch waren die Fehlerzahlen bei den Listen, die der Grammatik entsprachen geringer als bei den anderen Listen. Grund hierfür ist, dass das Netz die wahrscheinlichste Ausgabe produziert. Bei gängigeren Listen treten dadurch weniger Fehler auf als bei unwahrscheinlichen Reihenfolgen.

3.4.4 Bewertung des Modells

Mit dem Modell können Erklärungen für viele Phänomene gefunden werden, dabei auch für solche, die eigentlich als Argument gegen rekurrente neuronale Netze angeführt werden können.

Aus den Ergebnissen ihrer Arbeit ziehen Botvinick und Plaut zwei wichtige Folgerungen. Erstens, dass rekurrente Netzwerke wider alle Kritik in der Lage sind, die zentralen Leistungsergebnisse beim „serial recall“ nachstellen zu können und zweitens dass diese Netzwerke im Gegensatz zu dem kontextbasierten Modell auch Erklärungen für die Einflüsse von Hintergrundwissen bieten können (Botvinick & Plaut, 2006).

Als Ursache für den „list length“-Effekt beim „immediate serial recall“ nennt das Modell von Botvinick und Plaut (2006) zum einen den Einfluss des Zufalls beim Trainingsalgorithmus des Netzes und beim hinzugefügten Hintergrundrauschen, welcher in ihren Augen für Fehler des Modells beim Wiederholen von Listen verantwortlich ist. Ein weiterer Grund für den „list length“-Effekt stellt laut Botvinick und Plaut (2006) die Repräsentationsmethode des Netzes für eine gelernte Liste dar. Dadurch, dass die Liste als Überlagerung der Repräsentationen der einzelnen Listenelemente dargestellt wird, steigt die Aktivierung der verdeckten Schicht mit zunehmender Listenlänge.

Mit ansteigender Aktivierung nimmt gleichzeitig der Einfluss der zufälligen Abweichungen zu. Damit wird bewirkt, dass die Zerlegung der Überlagerung in die Repräsentationen der einzelnen Elemente zunehmend schwieriger wird (Botvinick & Plaut, 2006). Damit bewirkt eine zunehmende Listenlänge steigende Fehlerzahlen.

Zum „primacy“-Effekt und zum „recency“-Effekt tragen im Modell von Botvinick und Plaut (2006) mehrere Faktoren bei. Wie bereits erwähnt (vgl. Kapitel 3.4.2), ähneln sich die Repräsentationen von Listenelementen, die in der Liste benachbart sind. Die Elemente am Anfang sowie am Ende der Liste besitzen weniger benachbarte Elemente, als Listenelemente aus der Mitte der Liste. Somit existieren auch weniger Elemente, die über hohe Ähnlichkeit verfügen und dadurch leichter verwechselt werden können. Damit trägt eine Listenposition am Anfang oder am Ende einer Liste zu einer geringeren Fehlerzahl beim Wiederholen des Elementes bei (Botvinick & Plaut, 2006).

Als weiteren Grund für den „primacy“-Effekt nennen Botvinick und Plaut (2006), dass bei der Speicherung der ersten Listenelemente noch keine weiteren bzw. nur wenige Listenelemente gespeichert sind. Dadurch sind in der Gesamtzeit zwischen Speicherung und Wiederholung weniger andere Elemente kodiert. Da stärkere Überlagerungen zu höheren Fehlerzahlen führen, trägt die geringere Anzahl kodierter Elemente zu geringeren Fehlerzahlen der ersten Listenelemente bei. Da sich laut Botvinick und Plaut (2006) die Repräsentationen der bereits wiederholten Elemente von denen, die noch wiederholt werden müssen, unterscheiden lassen, trägt dieser Effekt in abgeschwächter Form auch zum „recency“-Effekt bei. Hier ist die Anzahl der noch zu wiederholenden Elemente für die Listenelemente zum Ende der Liste geringer.

Die beiden Fehlerarten der „errors of omission“ und der „errors of commission“ kann das Modell von Botvinick und Plaut (2006) durch die Repräsentation der einzelnen Listenelemente erklären. Dadurch, dass in einer Repräsentation sowohl das Element als auch seine Position innerhalb der Liste kodiert ist, können Fehler einerseits dazu führen, dass ein Element mit einem anderen verwechselt wird, und andererseits, dass eine andere Listenposition angenommen wird. In dem Fall, dass ein Element mit einem anderen verwechselt wird, entsteht somit eine „error of commission“ nachdem mit der korrekten Liste fortgefahren wird. Im anderen Fall, dass die aktuelle Listenposition mit einer späteren Position verwechselt wird, können Elemente, die eigentlich noch folgen müssten ausgelassen werden. Also wird ein „error of omission“ erzeugt.

Solche Fehler werden in den Simulationen von Botvinick und Plaut (2006) dadurch erzeugt, dass ein Rauschen in Form zufälliger Werte zur Aktivierung hinzuaddiert wird.

Meiner Meinung nach widerspricht das Prinzip der Informationsspeicherung durch Aktivierung aber den Erkenntnissen der Neurobiologie (siehe Abschnitt 4.1). Als Ursache für die Speicherung wurden Veränderungen der synaptischen Übertragungsstärke identifiziert (vgl. Kapitel 4.1). Diese werden im angegebenen Modell von Botvinick und Plaut (2006) jedoch durch einen Trainingsalgorithmus festgelegt und während der „serial recall“-Aufgabe nicht mehr verändert.

Diese Trainingsphase des neuronalen Netzes stellt eine weitere Eigenschaft dar, für die es keine Evidenz aus der Neurowissenschaft gibt. Beim Training werden die Übertragungsgewichte durch einen Algorithmus an die Aufgabenstellung angepasst, bis die gewünschte Leistungsfähigkeit erreicht ist. Im Modell von Botvinick und Plaut (2006) wird ein Backpropagation-Algorithmus für rekurrente neuronale Netze (Williams & Zipser, 1995) verwendet. Diese Art der überwachten Anpassung der Gewichte ist biologisch unplausibel (Garner, 2003).

Das Modell von Botvinick und Plaut (2006) wurde für die Aufgabe des „immediate serial recall“ entworfen. Dadurch kann es als einziges der vorgestellten Modelle auch problemlos Erklärungen für alle beschriebenen Effekte bei der Aufgabe des „immediate serial recall“ liefern. Da die Mechanismen der Informationsrepräsentation und des Trainings des Modells nach meiner Auffassung jedoch nicht in denen der Biologie wieder gefunden werden können, scheinen diese Erklärungen keine Aussagekraft über die Ursachen der Effekte zu besitzen.

Um alle beschriebenen Effekte zu erzeugen, benötigen Botvinick und Plaut eine große Anzahl verschiedener Mechanismen sowie eine komplexe Struktur. So wird beispielsweise eine Trainingsphase benötigt, damit das Netz überhaupt in der Lage ist, Sequenzen zu speichern. Das Netz besteht aus drei verschiedenen Schichten und zur Erzeugung der beschriebenen Fehler muss ein Hintergrundrauschen zu den Aktivierungen addiert werden. Meiner Ansicht nach resultieren die Effekte beim „immediate serial recall“ auf den grundlegenden Lernmechanismen und treten somit in den einfachsten Neuronenverbänden auf. Es sollte daher möglich sein, alle beschriebenen Effekte in einem Netz mit einfacherer Struktur und ohne vorherige Trainingsphase zu beschreiben. Dabei sollten die Fehler durch die Arbeitsweise des Mo-

dells erklärt werden können, und nicht durch die Addition von zufälligen Werten erzeugt werden.

3.5 Wichtige Probleme und Widersprüche aktueller Ansichten

Im vorherigen Kapitel habe ich einige aktuelle Modelle des Arbeitsgedächtnisses und seiner Mechanismen vorgestellt.

Die Modelle von Baddeley und Logie, Cowan sowie von Ericsson und Kintsch sind so allgemein und abstrakt formuliert, dass sie meiner Ansicht nach den Bezug zu den zugrunde liegenden Mechanismen verlieren. Somit werden Verhaltensformen von Testpersonen zwar durch die definierten Funktionen und Eigenschaften des Modells gedeutet, diese werden dabei aber nicht mit Funktionen oder Eigenschaften des menschlichen Gedächtnisses in Zusammenhang gebracht. Wenn beispielsweise das „multiple component model“ von Baddeley und Logie (1999) als Ursache für beeinträchtigtes Sprachenlernen eine gestörte Funktionalität des „phonological loop“ genannt wird, liefert das meines Erachtens keinen Hinweis darauf, welche biologischen Vorgänge die Ursache für dieses Verhalten sind, oder darüber hinaus wie eine solche Störung zu behandeln wäre.

Modelle des Arbeitsgedächtnisses werden konstruiert, weil in der Realität die verantwortlichen Prozesse noch nicht bekannt sind. Dabei sollten sie meiner Meinung nach jedoch die bereits erforschten Mechanismen nicht außer Acht lassen, sondern eine Art Bindeglied zwischen neurobiologischen Erkenntnissen und dokumentiertem Verhalten von Testpersonen bilden. Die Modelle von Baddeley und Logie, Cowan sowie von Ericsson und Kintsch stellen meiner Ansicht nach also nur oberflächliche Zusammenfassungen beobachteter Verhaltensformen dar, ohne diese begründen zu können. Es können zwar vage Vorhersagen getroffen werden, wie Reaktionen oder Versuchsergebnisse von Testpersonen aussehen, aber bei der Suche nach Ursachen für spezielle Verhaltensformen helfen die Modelle nur wenig. Um Erklärungen für neurologische Schädigungen oder kognitive Defizite zu finden und diese heilen oder behandeln zu können, ist das Verstehen konkreter Zusammenhänge der Mechanismen notwendig, was durch die gegebenen Modelle nicht ermöglicht wird. Die Effekte beim „immediate serial recall“ können die Modelle nur dadurch erklären, dass sie lückenhaft formuliert sind. Dadurch bleibt genug Freiraum für mögliche Mechanismen, die die beschriebenen Effekte erzeugen. Welche Mechanismen das speziell sind, erklären die Modelle nicht. Ein Beispiel dafür ist, dass weder Baddeley und Logie noch Cowan eine eindeutige Aussage treffen, ob Informationen im Arbeitsge-

dächtnis auf Grund von zeitlichem Verblässen oder auf Grund von Interferenzen nur für eine begrenzte Zeit gespeichert werden können.

Im Gegensatz zu diesen Modellen formulieren Botvinick und Plaut (2006) konkrete Angaben zu Mechanismen und Eigenschaften des Arbeitsgedächtnisses beim „serial recall“. Da diese aber meines Erachtens den Erkenntnissen der zellbiologischen Vorgänge im Gehirn widersprechen, stellt das Modell von Botvinick und Plaut ebenfalls keine Erklärung von Arbeitsgedächtnisprozessen dar. So stehen in dem Modell die Informationsspeicherung durch Aktivierung, die festen Übertragungsgewichte und die Trainingsphase des Netzes im Widerspruch zu den erforschten biologischen Prozessen.

Die vorgestellten Modelle bieten in unterschiedlichem Maße Erklärungen für einige oder auch alle vorgestellten Effekte, die bei der Aufgabe des „immediate serial recall“ auftreten. Teilweise werden hierbei die gleichen Mechanismen für einzelne Effekte verantwortlich gemacht, wie beispielsweise das Rehearsal für den „primacy“-Effekt im Modell von Baddeley und Logie (vgl. Kapitel 3.1.7) und im Modell von Cowan (vgl. Kapitel 3.2.5), teilweise widersprechen sich die Erklärungen jedoch. So argumentieren Baddeley und Logie (vgl. Kapitel 3.1) für eine konkrete Trennung akustischer und visueller Daten mit unterschiedlichen Funktionen für die Datentypen während Cowan (vgl. Kapitel 3.2) für einheitliche Funktionen ohne klare Trennung der Datentypen argumentiert.

Da die Erklärungen meiner Meinung nach aber wie erwähnt entweder nur sehr vage formuliert sind oder aber den biologischen Vorgängen widersprechen, besteht ein Bedarf nach einem Modell, das explizite Aussagen zu den Mechanismen des Arbeitsgedächtnisses auf der Basis des aktuellen Forschungsstands der Biologie liefert.

Im Folgenden werde ich ein Modell formulieren, das diesen Anforderungen entspricht.

4 Das Kanal-Modell des Arbeitsgedächtnisses

Ich definiere das Arbeitsgedächtnis in meinem Modell als die Struktur, die auf kurz oder langfristig gespeicherte Erfahrungen zugreifen kann und auf dieser Basis die passendste Reaktion zu einer gegebenen Situation berechnet.

Mein Modell lässt sich in seiner Grundidee dem Konnektionismus zuordnen. Viele Verarbeitungselemente sind miteinander verknüpft und das Wissen wird in den einzelnen Verbindungen und deren Muster gespeichert (Black, 1993).

Nach meiner Auffassung entspricht eine derartige Struktur mit verteilter Speicherung eher dem biologischen Vorbild, als beispielsweise Baumstrukturen mit expliziter Speicherung.

Da sich alle Mechanismen des Arbeitsgedächtnisses in den biochemischen Abläufen des Nervensystems wieder finden lassen müssen, bildet die Untersuchung der biologischen Zusammenhänge die Basis für die Entwicklung meines Modells.

4.1 Biologische Grundlagen des Modells

Als Ansatz für die Ermittlung der biologischen Grundlagen wähle ich die einfachsten biochemischen Reaktionen im Nervensystem auf zellulärer Ebene.

Schon in den Reaktionen einzelner Nervenzellen (*Neuronen*) lassen sich Mechanismen zur Kommunikation sowie zur temporären und dauerhaften Speicherung finden.

Diese Informationsspeicherungen finden sowohl auf genomischer, molekularer als auch auf zellulärer Ebene statt (Black, 1993; Pulves, Augustine, Fitzpatrick, Hall, LaMantia, McNamara, & Williams, 2004).

Informationen werden dabei durch Veränderungen von Struktur und Übertragungsfunktionen auf den unterschiedlichen Ebenen verarbeitet und gespeichert. Dabei tragen verschiedene körpereigene Moleküle zur Übertragung und Speicherung bei. Zwei davon sind *Neurotransmitter* und *Neurotrophine*. Neurotransmitter sind die Moleküle, die mit ihren entsprechenden Rezeptoren für die Informationsübertragung sorgen. Neurotrophine sind Moleküle, die zum Erhalt und zur Veränderung der zugrunde liegenden Struktur beitragen, indem sie von den Nervenzellen sowohl zum Überleben als auch zum Wachstum benötigt werden (Pinel, Pauli, 2007).

4.1.1 Struktur des Nervensystems

Das Nervensystem besteht aus einem stark verzweigten Netzwerk von Neuronen (Pulves et al., 2004). Von den Zellkörpern dieser Neuronen gehen verästelte Fortsätze aus. Diese unterscheiden sich in das *Axon*, das in so genannte Endknöpfchen endet und über diese mit benachbarten Nervenzellen verbunden ist und in die *Dendriten*, die meist kürzere Fortsätze darstellen, mit denen die Endknöpfchen anderer Nervenzellen verbunden sind (Pinel & Pauli, 2007). Die Verbindungsstellen der Endknöpfchen mit der Zellmembran einer weiteren Nervenzelle heißen *Synapsen*. An diesen Synapsen finden Signalübertragungen dadurch statt, indem aus den Endknöpfchen der präsynaptischen Zelle *Transmitterstoffe* in den synaptischen Spalt ausgeschüttet werden und *Rezeptormoleküle* in der Membran der postsynaptischen Zelle auf diese Transmitter reagieren. An der Oberfläche eines Neurons werden Signale über das vorliegende *Membranpotential* übertragen. Dieses Potential stellt eine momentan vorliegende elektrische Spannung an der Membran der Nervenzelle dar, welche durch unterschiedliche Ionenkonzentrationen hervorgerufen wird (Pinel & Pauli, 2007). Im Ruhezustand liegt an der Membran ein Ruhepotential von ca. -70 mV an, welches durch aktive, Energie verbrauchende Prozesse aufrechterhalten wird. Wird dieses Potential durch äußere Einflüsse verändert, so wird diese Veränderung durch die Zelle ähnlich einem elektrischen Kabel entlang der Membran weitergeleitet (Pinel & Pauli, 2007; Pulves et al., 2004). Äußere Einflüsse können beispielsweise Rezeptoren an Synapsen sein, die durch Verbindung mit Transmitterstoffen Ionenkanäle in der Membran öffnen.

Wenn das Membranpotential am Ursprung des Axons, dem *Axonhügel*, eine *Erregungsschwelle* übersteigt, wird dort durch spannungsgesteuerte Ionenkanäle das Membranpotential kurzzeitig auf eine Spannung von ungefähr +50 mV umgekehrt. Dieses *Aktionspotential* wird entlang des Axons zu den Endknöpfchen weitergeleitet. Dort verursacht es wiederum eine Ausschüttung der Transmitterstoffe in den synaptischen Spalt (Pinel & Pauli, 2007; Pulves et al., 2004). Aktionspotentiale sind dabei so genannte „Alles-oder-Nichts-Reaktionen“, deren Stärke auch bei unterschiedlich deutlicher Überschreitung der Erregerschwelle immer gleich ausfällt (Pinel & Pauli, 2007).

Sinneszellen können auf Umweltreize reagieren, indem sich ihr Membranpotential beispielsweise durch Licht- oder Hitzeeinfluss verändert.

Nervenzellen reagieren also zusammengefasst auf Potentialveränderungen an ihrer Membran, indem sie diese ab einer bestimmten Stärke an verknüpfte Neuronen weiterleiten. Genauere Vorgänge der Signalübertragung werden durch die Anzahl und Beschaffenheit der Transmitter und der Rezeptoren bestimmt.

4.1.2 Signalübertragung durch Transmitter und Rezeptoren

Neurotransmitter lassen sich in zwei Gruppen unterscheiden. Zum einen in die *niedermolekularen Neurotransmitter* und zum anderen in die *hochmolekularen Neurotransmitter*.

Niedermolekulare Neurotransmitter lassen sich in die Gruppen der *Aminosäuren*, der *Monoamine*, der löslichen Gase und des *Acetylcholins* einteilen.

Sie werden normalerweise direkt in den Synapsen synthetisiert und innerhalb der Zelle mit einer Membran umhüllt in so genannten *Vesikeln* zusammengefasst (Pinel & Pauli, 2007).

Alle hochmolekularen Neurotransmitter sind *Peptide*, d.h. Verbindungen aus mehreren Aminosäuren, und somit viel größer als die niedermolekularen Transmitter (Pinel & Pauli, 2007).

Sie werden im Gegensatz zu den Neurotransmittern im Zellkörper zusammengesetzt und anschließend in Vesikeln entlang des Axons bis zu den Synapsen transportiert. Veränderungen in der Synthese machen sich also erst nach einer gewissen Transportzeit bemerkbar (Abbildung 7).

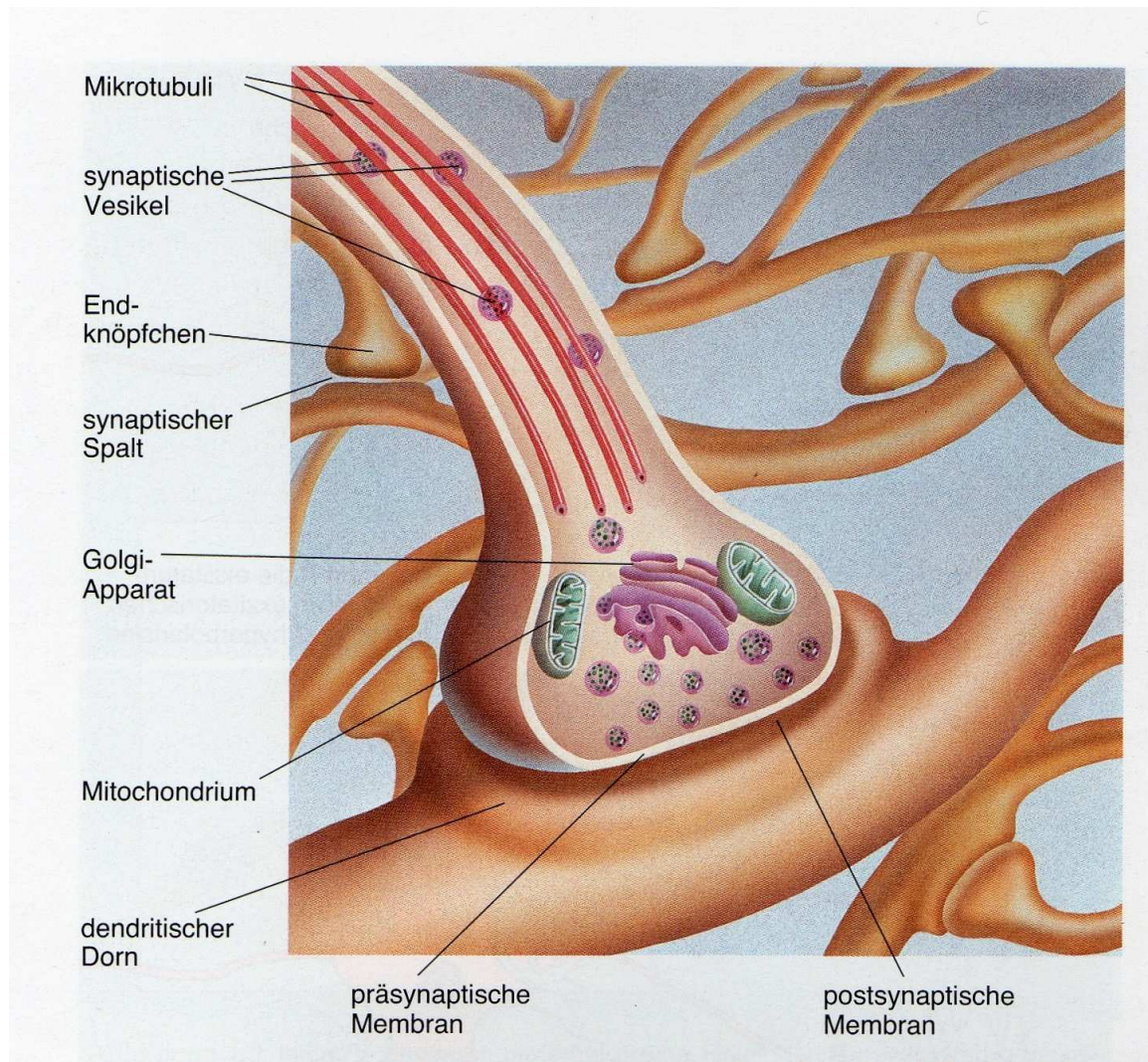


Abbildung 7: Anatomie einer Synapse (Pinel & Pauli, 2007)

Entsprechend der Transmittergröße sind auch die Vesikel mit Neuropeptiden größer und lagern sich nicht so dicht an der präsynaptischen Membran an (Pinel & Pauli, 2007).

Die Ausschüttung der niedermolekularen Transmitter findet dadurch schnell und bei jedem Aktionspotential statt, während die Ausschüttung der hochmolekularen Transmitter nur langsam, bei einer allgemeinen Zunahme der Feuerrate über eine bestimmte Frequenz, erfolgt (Black, 1993; Pinel & Pauli, 2007).

Niedermolekulare und hochmolekulare Transmitter koexistieren in Neuronen (Black, 1993; Pinel & Pauli, 2007) und können unabhängig voneinander auf Reize reagieren. Die Anzahl und das Verhältnis der Transmitter können sich durch Umwelteinflüsse verändern und stellen eine Art Zustand eines Neurons dar.

Rezeptoren für die Transmitterstoffe befinden sich hauptsächlich auf der Postsynapse aber auch vereinzelt auf der Präsynapse. Die Rezeptoren auf der Präsynapse

regulieren im Allgemeinen die Transmitterausschüttung. So reagieren manche auf den ausgeschütteten Transmitter der Präsynapse selbst und andere auf diffus verbreitete Transmitter wie beispielsweise Hormone. So kann das Hormonsystem auf die Reizverarbeitung einwirken. Diese Kommunikation ist nicht auf eine direkte Verschaltung angewiesen. Der restliche Körper trägt also zur Informationsverarbeitung bei (Black, 1993; Pinel & Pauli, 2007).

Es existieren zwei Arten von Rezeptoren, *ionotrope Rezeptoren* und *metabotrope Rezeptoren*. Ionotrope Rezeptoren öffnen nach Kopplung des Transmitters bestimmte Ionenkanäle zur De- oder Hyperpolarisation der Nervenzelle. Metabotrope Rezeptoren geben nach Bindung mit dem Transmitter ein so genanntes *G-Protein* ab. Dieses kann wiederum auch einen Ionenkanal öffnen oder die Synthese eines *sekundären Botenstoffes* bewirken. Sekundäre Botenstoffe können sich beispielsweise an Moleküle binden, welche die RNA-Synthese bestimmter Gene verstärken (Pulves et al., 2004). Dieser Veränderung der Genexpression kann lang anhaltende Folgen, wie die Neubildung von Synapsen, haben (Pinel & Pauli, 2007; Pulves et al., 2004).

Niedermolekulare Neurotransmitter werden eher direkt an Synapsen ausgeschüttet und beeinflussen ionotrope oder metabotrope Rezeptoren, die direkt Ionenkanäle beeinflussen. Hochmolekulare Neurotransmitter hingegen werden eher diffus eingesetzt und wirken verstärkt auf metabotrope Rezeptoren, die sekundäre Botenstoffe abgeben (Pinel & Pauli, 2007).

Damit Transmitter nicht dauerhaft für die Öffnung von Ionenkanälen sorgen werden sie sowohl durch die Präsynapse wieder aufgenommen als auch durch Enzyme im synaptischen Spalt zerlegt (Pinel & Pauli, 2007; Pulves et al., 2004).

4.1.3 Verschiedene Mechanismen zur Informationsspeicherung

Umweltreize von Sekunden oder Minuten können Transmitteränderungen von Tagen oder Wochen bewirken (Black, 1993).

Ein Beispiel für solche Effekte zeigt sich in der Signalübertragung durch *Katecholamine*. Katecholamine gehören zur Gruppe der Monoamine unter den niedermolekularen Neurotransmittern. Zu ihnen gehören unter anderem *Adrenalin* und *Noradrenalin*. Katecholamine hemmen innerhalb der Nervenzelle die Aktivität des Enzyms, welches für ihre Synthese zuständig ist. Eine Ausschüttung der Katecholamine in den synaptischen Spalt bewirkt damit eine Enthemmung und somit eine kurzzeitig erhöhte Transmittersynthese (Black, 1993). Direkt folgende Signale werden dadurch stärker übertragen.

Nach Einstrom von Kalziumionen in die präsynaptische Zelle benötigt diese einige Zeit, um die ursprüngliche Konzentration wiederherzustellen, d.h. dass die Folgen einer Aktivierung kurzzeitig erkennbar bleiben (Pulves et al., 2004). Aufeinanderfolgende Reize addieren sich auf diese Weise zu einem höheren Membranpotential. Diese Erhöhung hält nur für wenige Millisekunden an.

Halten die Reize an, wird die in den Vesikeln verfügbare Transmittermenge aufgebraucht und die Übertragungsstärke der Synapse nimmt ab.

Solche Mechanismen stellen, wenn auch nur für eine sehr kurze Zeit, eine Speicherung von Informationen dar.

Viele hochfrequent aufeinander folgende Reize (= *Tetanus*) können eine so genannte *post-tetanic potentiation* (PTP) hervorrufen, die auch noch wenige Minuten nach dem ersten Reiz eine erhöhte Reaktion auf eine Stimulierung darstellt (Pulves et al., 2004). Diese resultiert wahrscheinlich aus der Aktivierung präsynaptischer Kinasen, die die Transmitterausschüttung der Vesikel erleichtern.

Diese Mechanismen wurden an praktisch allen Synapsen beobachtet, die genauen Mechanismen sind jedoch noch nicht bekannt.

Ein Beispiel für eine länger anhaltende Auswirkung eines Aktionspotentials an einer Synapse stellt die Phosphorylierung eines Enzyms durch einen sekundären Botenstoff dar, das dadurch für bis zu 30 Minuten verstärkt Katecholamine synthetisiert (Black, 1993; Pulves et al. 2004).

Eine weitere Art der längerfristigen Speicherung stellt die long-term potentiation (LTP) dar. Die LTP zeigt bei intensiven transsynaptischen Übertragungen eine Verstärkung dieser Synapsen, die bis zu vielen Wochen anhalten kann. Sie wurde in „[...] vielen Spezies und in vielen Teilen ihrer Gehirne nachgewiesen aber am häufigsten wurde sie im Hippocampus der Ratte untersucht“ (Pinel & Pauli, 2001, S. 371).

Bei der LTP im Hippocampus wurde ein großer Einfluss von so genannten NMDA (N-Methyl-D-Aspartat)-Rezeptoren der postsynaptischen Zelle nachgewiesen. Diese bewirken in Verbindung mit dem Neurotransmitter Glutamat nur einen geringen Einstrom von Kalziumionen in das postsynaptische Neuron, da sie durch das Membranpotential mittels eines positiv geladenen Magnesiumions von außen verstopft wird. Ist die postsynaptische Zelle jedoch depolarisiert, löst sich die Verstopfung durch das Magnesiumion, und Kalziumionen können in die Nervenzelle einströmen.

Kalzium wirkt in der Nervenzelle als sekundärer Botenstoff und bewirkt wahrscheinlich die Synthese zusätzlicher Rezeptoren an der Synapse durch Aktivierung einer Proteinkinase (Pulves et al., 2004; Sweatt, 2003).

Langanhaltende schwache Reize führen wiederum zu einer long-term depression (LTD) und einer Verminderung der Rezeptoren (Pinel & Pauli, 2007; Sweatt, 2003).

Länger andauernde LTP führt neben der Zunahme der Rezeptorenanzahl zu strukturellen Veränderungen und Vermehrungen der entsprechenden Synapsen und der dendritischen Dornen. Als Ursache hierfür wird vermutet, dass Kalziumionen in der postsynaptischen Zelle über sekundäre Botenstoffe auch zu verstärkten Genexpressionen führen (Black, 1993; Pulves et al., 2004; Sweatt, 2003). Sowohl die LTP kann durch ähnliche Prozesse wie bei der LTD rückgängig gemacht werden, als auch die LTD durch vergleichbare Prozesse wie bei der LTP (Sweatt, 2003).

Eine weitere Form der Informationsspeicherung in quasi permanenter Form stellt die generelle Struktur der Verknüpfung des Nervensystems an sich dar. Körpereigene Moleküle steuern während der Entwicklung Synapsen- und Axonwachstum und auch den späteren Erhalt (Black, 1993).

Hier findet eine ständige Synapsenneuordnung in Folge von Neuronentod statt. Beim Neuronentod konkurrieren Nervenzellen um notwendige Substanzen. Neuronen, deren Synapsen sich auf eine größere Anzahl verschiedener Nervenzellen verteilen, unterliegen hierbei und werden anschließend durch Zerlegungsprozesse aufgelöst. Die Synapsen von neu gebildeten Neuronen verteilen sich weniger stark auf unterschiedliche Nervenzellen. So nimmt die Selektivität der Übertragung zu (Pinel & Pauli, 2007).

4.1.4 Entwicklung unterschiedlicher Areale im Gehirn

Während der frühen Entwicklungsphase des zentralen Nervensystems sorgen Neurotrophine dafür, dass in bestimmten Regionen bestimmte Arten von Nervenzellen gebildet werden. Diese unterscheiden sich von den Nervenzellen in anderen Regionen sowohl durch die Verknüpfung als auch durch die Transmitter, die sie ausschütten. Beim Axonwachstum und der Synapsenbildung bewegt sich der Wachstumskegel auch über größere Distanzen hinweg zu den Nervenzellen, mit denen er eine Verbindung eingehen muss (Pinel & Pauli, 2007; Pulves et al., 2004). Hierbei erfolgt die Richtungsfindung ebenfalls über die Anwesenheit einer speziellen Konzentration von Molekülen, die das Wachstum in die entsprechende Richtung begünstigen. So bilden sich unterschiedlich spezialisierte Gehirnregionen aus, die über fest

vorgeschriebene Verbindungen zu anderen Regionen verfügen. Bildgebende Verfahren zeigten, dass diesen Gehirnregionen unterschiedliche Aufgabenbereiche zugeordnet werden können.

Diese Regionen unterscheiden sich in den Transmitterstoffen und Transmitterkonzentrationen, die synthetisiert werden (Black, 1993).

4.1.5 Zusammenfassung

Das Nervensystem besteht aus einem Netzwerk von einzelnen Neuronen. Signale werden von einer Nervenzelle nur weitergegeben, wenn sie eine Erregerschwelle überschreiten. Die Übertragung zwischen zwei Neuronen findet in der Regel an Synapsen statt. Die Übertragungsstärke einer Synapse kann durch eine unterschiedliche Anzahl an Transmittern oder Rezeptoren variieren.

Dabei bewirken leichte und nur kurzzeitig andauernde Signale eher die Ausschüttung von niedermolekularen Transmittern. Hochmolekulare Transmitter werden nur bei stärkeren oder länger andauernden Signalen zusätzlich abgegeben.

Niedermolekulare Transmitter bewirken in der Regel nur die Öffnung von Ionenkanälen. Gesteigerte Syntheseraten und erleichterte Transmitterausschüttung nach der post-tetanic Potentiation bewirken eine sofortige Erhöhung der Übertragungsstärke für Millisekunden, bis zu wenigen Minuten nach einer Aktivierung normalisieren sich aber schnell wieder.

Somit wirken sich die niedermolekularen Neurotransmitter in der Regel schnell aus, können Informationen aber nur kurzzeitig speichern.

Neuropeptide bewirken meist durch sekundäre Botenstoffe die Aktivierung von Enzymen, was zu unmittelbarer Proteinsynthese oder sogar zu verstärkter Genexpression führen kann. Somit machen sich die Auswirkungen hochmolekularer Transmitter zwar später bemerkbar, können aber zu mittel- bis langfristigen Veränderungen von Synapsenstärken führen. Die Speicherdauer bewegt sich im Bereich von 30 Minuten bei Enzymaktivierung bis zu quasi-permanenten Strukturänderungen nach verstärkter Genexpression.

Diese Grundprinzipien stellen eine rudimentäre Form des Gedächtnisses für unterschiedliche Dauer dar und sind in einfachsten Neuronenansammlungen zu finden. Selbst bei entfernt verwandten Organismen lassen sich die gleiche Mikroarchitektur und die gleichen neuronalen Prozesse finden (Black, 1993).

Das Nervensystem ist entwicklungsbedingt strukturell in Bereiche eingeteilt, die sich in der Art der Daten unterscheiden, die sie verarbeiten.

4.2 Struktur und Mechanismen des Modells auf Basis der biologischen Grundlagen

4.2.1 Struktur

Mein Modell des Arbeitsgedächtnisses stellt ein rekurrentes neuronales Netzwerk aus einzelnen Verarbeitungselementen (*Units*) dar. Dieses entspricht dem Vorbild des Netzwerkes aus Nervenzellen, wobei die Erregerschwelle durch einen Schwellenwert der Units dargestellt wird.

Verschiedene Übertragungsstärken von Synapsen werden im Modell durch Übertragungsgewichte zwischen den Units realisiert. Diese Übertragungsgewichte teilen sich in Kurzzeit- und Langzeit-Gewichte, basierend auf niedermolekularen und hochmolekularen Transmittern. Für jede Verbindung existieren theoretisch also zwei unabhängige Gewichte, die sich in unterschiedlicher Form und Dauer anpassen können. Negative Werte für Gewichte entsprechen dabei einer hemmenden Funktion.

Die Gewichte erfahren eine Veränderung, je nachdem ob sie bei einer Übertragung eines Signals zur Aktivierung der folgenden Unit beitragen, oder ob die nachfolgende Unit nicht aktiviert wird obwohl sie ein Signal überträgt.

Die Menge der Units lässt sich entsprechend der Aufteilung des Nervensystems in einzelne Bereiche in einen Kanal und einen Kontext trennen. Der Kanal stellt dabei die Bereiche des Nervensystems dar, die an der aktuellen Aufgabe beteiligt sind, und der Kontext alle restlichen Bereiche. Dieses spiegelt in gewisser Weise die einzelnen Komponenten im „Multiple-Component Modell“ von Baddeley und Logie (vgl. Kapitel 3.1), wobei sich der Kanal in meinem Modell nicht auf eine bestimmte Art von Reiz (akustisch oder visuell) oder einen einzelnen Bereich des Gehirns („phonological loop“ oder „visuospatial sketchpad“) beschränkt, sondern auch eine Kombination von mehreren unterschiedlichen Reizen und Bereichen darstellen kann.

4.2.2 Mechanismen

Repräsentation:

Die Repräsentation einer Information besteht in meinem Modell aus dem zusammengefassten Aktivierungsmuster der Units des Kanals, die durch den zugehörigen Reiz hervorgerufen worden. Dies ähnelt der Sichtweise von Cowan (vgl. Kapitel 3.2), der von einer aktivierten Teilmenge des Langzeitgedächtnisses ausgeht. Das Biolo-

gische Vorbild für aktivierte Units sind Neuronen, an denen ein Aktionspotential vorliegt.

Zusammenhänge zwischen verschiedenen Informationen werden über unterschiedlich stark ausgeprägte Verbindungen verkörpert. Verschiedene Verbindungsstärken entsprechen hierbei unterschiedlich starken Synapsen.

Speicherung:

Die Ebenen der Speicherung teilen sich nicht nur in Kurzzeit- und Langzeitwissen, sondern in Beziehung auf das biologische Vorbild in mehrere unterschiedlich lang andauernde Repräsentationsebenen.

Somit lassen sich die Ebenen der konnektionistischen Speicher- und Verarbeitungsstruktur in kurz- und langfristige einteilen.

Aufgrund der molekularen Mechanismen, die den unterschiedlichen Speichermetoden unterliegen, setzt eine langfristige Speicherung immer auch alle kürzer andauernden Speicherungen voraus.

Die Lernregel von Hebb (1949), dass eine gemeinsame Aktivierung zu einer Verstärkung der Verbindung führt, wurde durch die biologischen Vorgänge bestätigt, dass nur eine transsynaptische Übertragung zu einer LTP führen kann (vgl. Kapitel 4.1.3). Somit wird die Verbindung zwischen Unit A und Unit B nur gestärkt, wenn auf die Aktivierung von Unit A unmittelbar eine Aktivierung von Unit B folgt. Wenn Unit A aktiviert ist, Unit B aber inaktiv bleibt, findet eine Schwächung der Übertragungswirkung statt.

Diese Veränderungen der langfristigen Übertragungsstärke basiert auf den Transmitterzahlen der hochmolekularen Neurotransmitter. Sie finden nur nach besonders hochfrequenten Reizübertragungen statt und wirken sich durch die längeren Transportwege der Transmitter nur träge aus.

Veränderungen in der Übertragungsstärke auf Basis der niedermolekularen Transmitter finden hingegen schon nach einer alleinigen Ausschüttung der Präsynapse statt, unabhängig von einer Aktivierung der postsynaptischen Zelle.

Wenn diese hochfrequenten Reizübertragungen in stärkerer oder häufigerer Form stattfinden, würden dann die Prozesse ablaufen, die zu Neubildung von Synapsen führen, und somit eine quasi-permanenten Speicherung darstellen. Diese quasi-permanenten Veränderungen entsprächen einer Veränderung in der Struktur. Diese Strukturveränderungen werden in meinem Modell vernachlässigt, da es den Rahmen dieser Diplomarbeit sprengen würde.

Erhalt der gespeicherten Daten:

Neue oder widersprüchliche Beziehungen von Informationen führen dazu, dass vorher gelernte Zusammenhänge durch neue Zusammenhänge überschrieben werden. So wird beispielsweise eine vorherige Stärkung eines Übertragungsgewichtes durch eine anschließende Schwächung unwiderruflich überschrieben. Dies entspricht dem Prinzip der Interferenz.

Kurz- und langfristige Speichermechanismen können dadurch nach und nach immer mehr überschrieben werden, bis die gespeicherten Informationen nicht mehr verfügbar sind. Findet innerhalb der Speicherdauer eine wiederholte Übertragung an den Gewichten statt, werden die Veränderungen erneuert. Dieser Vorgang kann mit einer Art Rehearsal verglichen werden, wie er von Baddeley und Logie beschrieben wird (vgl. Kapitel 3.1).

Abrufen der gespeicherten Informationen:

Dadurch, dass Zusammenhänge von Informationen in den Übertragungsgewichten der Struktur gespeichert sind, können sie nur aktiviert werden, wenn der vorangehende Zustand zumindest teilweise, vorher aktiviert ist.

Hierbei spielen die einzelnen Übertragungsstärken eine Rolle dabei, welche Units an der Reaktivierung der gesuchten Information einen größeren Beitrag leisten als andere. So können wenige aktivierte Units mit starken Übertragungsgewichten genauso hinreichend sein, wie viele aktivierte Units mit nur schwachen Übertragungsgewichten. Dabei spielt es keine Rolle, ob der vorangehende Zustand durch innere Verarbeitungsprozesse erzeugt oder durch äußere Reize hervorgerufen wird.

Die Berechnung eines Folgezustandes des neuronalen Netzes basiert auf dessen Struktur. In dieser ist die quasi-permanente Speicherung repräsentiert. Somit trägt Langzeitwissen, welches quasi-permanent gespeichert ist, direkt zu der Verarbeitung des Arbeitsgedächtnisses bei. Dies entspricht der Vorstellung von „Long Term-Working Memory“ aus dem Modell von Ericsson und Kintsch (vgl. Kapitel 3.3).

Steuerung:

Die Lösungsfindung des Gedächtnisses für eine gestellte Aufgabe findet durch passive Prozesse statt, die verteilt durch die einzelnen Mechanismen innerhalb der Neuronen gesteuert werden. So rufen Reize eine interne Repräsentation hervor und die verbundenen Nervenzellen werden entsprechend ihrer Übertragungsstärken aktiviert. Wenn eine Lösung zu dem aktuellen Problem bekannt ist, wird die Repräsentation

tion dieser Lösung dadurch aktiviert. Gegebenenfalls werden Hinweise oder Zusatzinformationen benötigt, damit diese Lösung gefunden werden kann.

Es existiert also meines Erachtens keine zentrale Steuereinheit, die die Informationsverarbeitung lenkt, wie es in den Modellen von Baddeley und Logie (vgl. Kapitel 3.1), Cowan (vgl. Kapitel 3.2) sowie Ericsson und Delaney (vgl. Kapitel 3.3) der Fall ist.

4.3 Umsetzung des Modells als neuronales Netz für die Aufgabe des „immediate serial recall“

Die im vorhergehenden Kapitel beschriebene konnektionistische Struktur habe ich in ein implementiertes neuronales Netz für die Aufgabe des „immediate serial recall“ umgesetzt. Dabei wurden an verschiedenen Stellen Vereinfachungen gegenüber dem theoretischen Modell durchgeführt, um zum einen den Rahmen der Arbeit nicht zu sprengen und zum anderen Aspekte, die meiner Ansicht nach für die gewählte Aufgabe nicht notwendig sind, einzusparen. So beschränke ich mich beim „immediate serial recall“ auf Aufgaben, bei denen Einflüsse von quasi-permanent gespeicherten Erfahrungen ausgeschlossen werden. Im modellierten neuronalen Netz finden somit nur die Kurzzeit- und Langzeitveränderungen von Übertragungsgewichten statt.

4.3.1 Aufbau des neuronalen Netzes

Das Netz besteht aus einer einzelnen Schicht, deren Units mit allen Units der Schicht verbunden sind. Einzelne Units repräsentieren dabei Zusammenschlüsse mehrerer verbundener Neuronen im biologischen Vorbild. Somit können Units des Netzes auch eine Verbindung zu sich selbst besitzen.

Die Units der Schicht unterscheiden sich in einen Kanal und einen Kontext (vgl. Kapitel 4.2.1). Der Anteil des Kanals an der gesamten Schicht, wie auch die Schichtgröße, können für unterschiedliche „immediate serial recall“-Aufgaben variiert werden.

Die verschiedenen Übertragungsgewichte für die kurzzeitigen und langfristigen Veränderungen der Übertragungsstärken werden durch zwei separate Gewichtsmatrizen gleicher Größe dargestellt. Der Einfluss von quasi-permanent gespeicherten Informationen wird somit im Modell ausgeschlossen. Dabei entspricht ein Wert einer Gewichtsmatrix an der Stelle (x,y) dem Übertragungsgewicht von Unit x zu Unit y . Diese Übertragungsgewichte können nur die Werte -1 , 0 oder 1 annehmen. Dadurch werden die Verbindungen nur in verstärkte Verknüpfungen und nicht verstärkte Verknüpfungen unterschieden. Diese Vereinfachung wird angewandt, da sich zum einen genaue Werte für den Faktor einer Verstärkung nicht aus den Transmitterzahlen von Synapsen ableiten lassen und für die Funktionsweise des Netzes zum anderen auch nicht benötigt werden. Der Wert -1 stellt eine verstärkt hemmende Übertragung und der Wert 1 eine verstärkt erregende Übertragung dar. Übertragungsgewichte mit dem

Wert 0 stellen Verbindungen dar, an denen keine Veränderung der Übertragungsstärke stattgefunden hat. Diese Verbindungen sind dadurch in so schwacher Form vorhanden, dass sie im Verhältnis zu den verstärkten Gewichten vernachlässigbar sind.

Vor Beginn der „immediate serial recall“-Aufgabe besitzen alle Gewichte den Wert 0. Die Units des Netzes an sich können zwei unterschiedliche Zustände besitzen. Entweder sind sie zum aktuellen Zeitpunkt aktiviert, dann besitzen sie den Wert 1, oder sie sind nicht aktiviert und besitzen den Wert 0.

4.3.2 Input des Netzes

In meinem theoretischen Modell verursachen die entsprechenden Sinnesorgane entsprechend ihrer Verknüpfungen mit dem Netzwerk ein Aktivierungsmuster, das den aufgenommenen Reiz repräsentiert. Aus Gründen der Vereinfachung bilde ich diesen Prozess dadurch ab, dass ein Reiz direkt zur Aktivierung der jeweiligen Repräsentation im zugehörigen Kanal der Schicht führt.

Eine solche Repräsentation eines Reizes entspricht einem Binärstring mit der Größe des Kanals. Eine 1 an der Stelle x im Binärstring bewirkt eine Aktivierung der Unit x des Kanals. Der Anteil der Aktivierung, die ein Reiz hervorruft, kann vor der Aufgabe eingestellt werden.

Die Eingabe des Netzes besteht neben der Eingabe für den Kanal aus einer Eingabe für den Kontext. Eingaben für den Kanal stellen dabei Repräsentationen von möglichen Listenelementen dar, die vor einer „serial recall“-Aufgabe generiert werden. Die Eingaben für den Kontext werden zufällig zum Zeitpunkt der Präsentation eines Listenelementes erzeugt und besitzen keine besondere Bedeutung, die ihnen zugeordnet werden kann.

Vor Beginn der Präsentation wird eine spezielle Eingabe getätigt, die den Anfang der Aufgabe repräsentiert. Nach der Eingabe des letzten Listenelements wird wiederum eine spezielle Eingabe als Repräsentation des Endes der Liste präsentiert. Diese beiden Eingaben rufen ausschließlich Aktivierungen im Kontext hervor, um darzustellen, dass sie keinem Listenelement entsprechen.

4.3.3 Berechnung eines Folgeschrittes

Die nachfolgende Aktivierung des Netzes wird dadurch berechnet, dass für jede Unit die Summe aller Übertragungsgewichte, die von aktivierten Units ausgehen, ge-

bildet wird. Wenn diese Summe einen Schwellenwert überschreitet wird die entsprechende Unit ebenfalls aktiviert.

$$a(x, t) = 1 \quad \text{wenn } (W_1 + W_2) * A(N, t - 1) - B$$

an der Stelle $x > 0$

$$a(x, t) = 0 \quad \text{sonst}$$

$a(x, t)$ stellt dabei die Aktivierung einer Unit x zum Zeitpunkt t dar, die auf die Aktivierung des Netzes $A(N, t-1)$ zum Zeitpunkt $t-1$ folgt. W_1 und W_2 repräsentieren die Matrizen für die beiden Übertragungsgewichte.

Der Schwellenwert B berechnet sich aus dem Durchschnitt der Summe aus dem Produkt der erregenden Aktivierung für alle Units zum vorherigen Zeitpunkt und der Summe der Gewichtsmatrizen. Er ist dadurch für jede Unit gleich.

$$B = \frac{\sum_{i=1}^n V(i)}{n} \quad \text{mit } V = (W_1 + W_2) * A(N, t - 1)$$

und $V(i) = 0$ wenn $V(i) < 0$

Wobei n die Anzahl der Units und V den Vektor der Größe n mit dem Produkt aus vorheriger Aktivierung und den Gewichtsmatrizen darstellt.

Während der Lernphase der Liste wird mit den beschriebenen Funktionen die erwartete Aktivierung des Netzes berechnet. Die tatsächlich stattfindende Aktivierung entspricht aber dem Aktivierungsmuster des Listenelementes und dem aktuellen zufälligen Zustand des Kontextes. Das heißt, dass die Präsentation eines Listenelementes dadurch simuliert wird, dass im Kanal, unabhängig von vorhandenen Übertragungsgewichten, die Repräsentation des aktuellen Elementes als Aktivierungsmuster erzeugt wird. Die berechnete Aktivierung dient dabei der Lernregel für die kurzfristigen Gewichte um die Abweichung zwischen Vorhersage und tatsächlicher Aktivierung festzustellen (vgl. Kapitel 4.3.4).

4.3.4 Lernfunktionen

Entsprechend der beschriebenen Mechanismen des Modells (vgl. Kapitel 4.2.2) bestehen für die beiden Arten von Übertragungsgewichten unterschiedliche Lernfunktionen.

Die Änderung der Übertragungsgewichte der zweiten Matrix, die für die längerfristige Informationsspeicherung zuständig ist, findet entsprechend dem biologischen Vorbild nur nach starken Reizen statt. Solche starken Reize können in meinem Modell nur durch äußere Einflüsse erzeugt werden. Somit verändern sich die Werte der Übertragungsgewichte nur, wenn eine Eingabe stattgefunden hat. Gewichte die von einer aktivierten Unit zu einer ebenfalls aktivierten Unit führen werden auf den Wert 1 gesetzt (siehe Fall *a*) in Abbildung 8) und Gewichte, die von einer aktivierten Unit zu einer nicht aktivierten Unit führen auf den Wert -1 (siehe Fall *c*) in Abbildung 8). Dies entspricht den Lernregeln von Hebb (1949). Wenn eine Unit nicht aktiviert war, werden die Gewichte, die von dieser Unit zu einer anderen Unit führen nicht verändert, unabhängig davon, ob die andere Unit aktiviert wird oder nicht (siehe Fälle *b*) und *d*) in Abbildung 8). Die beschriebene Trägheit der Veränderungen der Übertragungsgewichte für längerfristige Speicherung (vgl. Kapitel 4.2.2) wird im neuronalen Netz dadurch simuliert, dass Übertragungsgewichte, die einmal auf den Wert 1 gesetzt wurden diesen auch bis zum Ende der Aufgabe behalten.

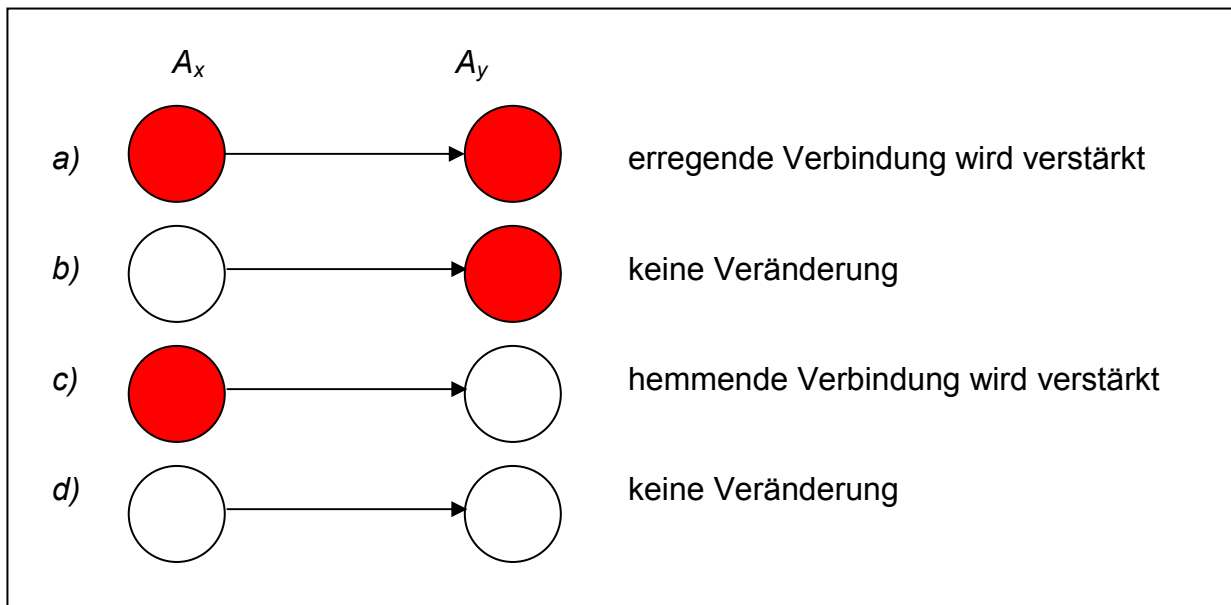


Abbildung 8: Lernregel für die Übertragungsgewichte. Aktivierte Neuronen sind rot dargestellt während inaktive Neuronen weiß sind.

Eine Veränderung eines Übertragungsgewichtes der ersten Matrix, die die kurzzeitige Speicherung repräsentiert, findet statt, wenn die Aktivierung der Schicht einer vorausberechneten Erwartungsaktivierung widerspricht. Wird die Aktivierung einer Unit erwartet, eine Eingabe bewirkt jedoch keine Aktivierung der Unit, führt dies dazu, dass alle Verbindungen von den vorher aktivierten Units auf den Wert -1 gesetzt werden. Wird jedoch erwartet, dass eine Unit nicht aktiviert wird, diese Aktivierung

jedoch durch eine Eingabe hervorgerufen wird, werden die Verbindungen von den vorher aktivierten Units auf den Wert 1 gesetzt. Ein derartiger Widerspruch zwischen einer erwarteten Aktivierung und einer statt findenden Aktivierung kann nur während der Lernphase entstehen, da während der Wiederholungsphase die erwartete Aktivierung der tatsächlich statt findenden Aktivierung entspricht.

Diese Gewichtsveränderungen entsprechen dem Vorbild der Erhöhung der Transmittermenge nach einer Ausschüttung. Wird eine Unit also unerwarteterweise aktiviert, hat an den erregenden Synapsen eine Ausschüttung statt gefunden und die Transmitterzahl wird erhöht (siehe Fall *a*) in Abbildung 8. Dem entsprechend führt das Ausbleiben einer erwarteten Aktivierung zu einer Erhöhung der Transmitterzahl an den hemmenden Synapsen (siehe Fall *c*) in Abbildung 8. Wenn sich tatsächliche Aktivierung und Erwartung gleichen, findet keine Veränderung der Übertragungsgewichte statt (siehe Fälle *b*) und *c*) in Abbildung 9).

Im Gegensatz zu den Langzeit-Gewichten benötigen die Kurzzeit-Gewichte keine starken Reize, die nur durch einen Input erzeugt werden können. Sollten also interne Mechanismen beispielsweise aus lückenhaften Repräsentationen ein Objekt erkennen und anschließend die vollständige Repräsentation aktivieren, würde ebenfalls eine Differenz zwischen Erwartung und tatsächlicher Aktivierung zu Lernprozessen führen. Diese inneren Mechanismen könnten beispielsweise durch eine zusätzliche Schicht umgesetzt werden.

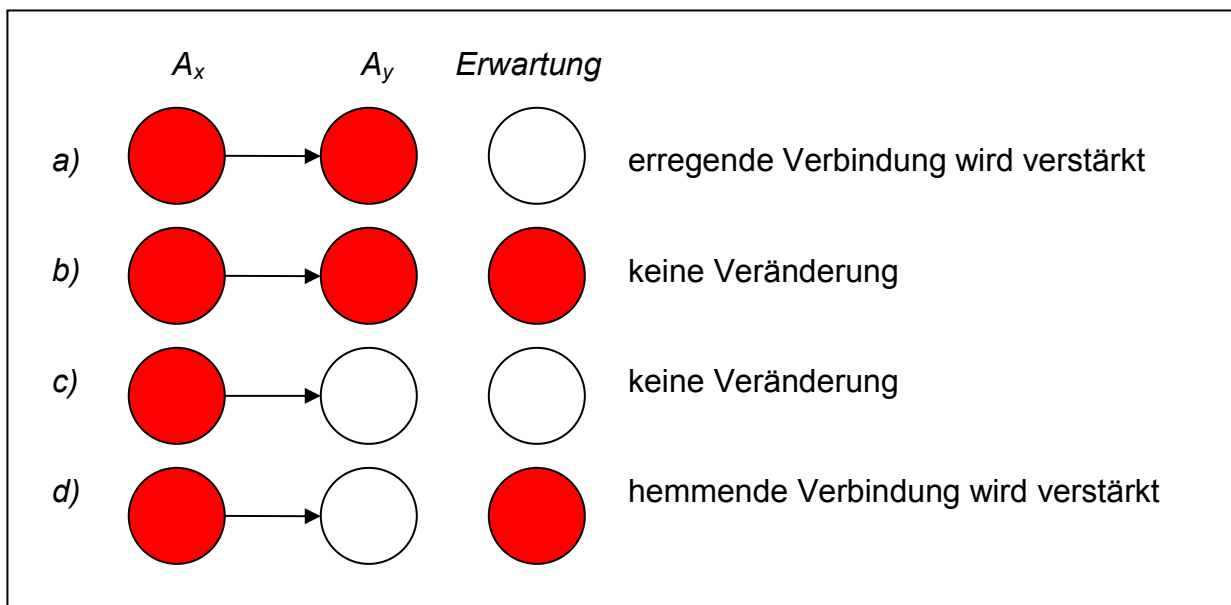


Abbildung 9: Zusätzliche Unterscheidung für die kurzfristigen Übertragungsgewichte.

4.3.5 Output des Netzes

Vor Beginn der Aufgabe wird eine Menge von zufälligen Repräsentationen von möglichen Listenelementen generiert. Während der Wiederholungsphase werden die einzelnen Aktivierungsmuster des Kanals mit den Elementen dieser generierten Menge verglichen. Das Element, das auf Basis der *Hamming-Distanz* die größte Ähnlichkeit mit der aktuellen Aktivierung besitzt, wird als Ausgabe des Netzes erkannt. Die Hamming-Distanz berechnet dabei die Anzahl der Stellen, an denen sich jeweils zwei Binärstrings unterscheiden.

Das Netz produziert somit nur Ausgaben, die in der Menge der möglichen Eingaben vorhanden sind.

4.3.6 Zusammenspiel der Komponenten

Zu Beginn eines Durchlaufs der „immediate serial recall“-Aufgabe durch das Modell besitzen sowohl alle Gewichte als auch alle Units der Schicht den Wert 0. Vor der Präsentation der Liste wird zuerst eine Menge mit möglichen Listenelementen generiert. Diese entsprechen Vektoren deren Größe der Größe des Kanals gleicht. Anschließend werden zufällige Repräsentationen des Starts und des Endes der Aufgabe erzeugt. Die Repräsentation des Aufgabenstarts wird in der Schicht aktiviert. Anschließend werden nacheinander die einzelnen Listenelemente präsentiert. Dabei lernen die langzeit-Gewichte anhand der momentanen Aktivierung und der Aktivierung, die dem neuen Listenelement entspricht, dass gerade präsentiert wird. Bei den kurzzeit-Gewichten wird gleichzeitig mit den momentanen Gewichten die Erwartung berechnet und mit dieser Erwartung, wie in Kapitel 4.3.4 beschrieben, gelernt. Parallel zu den Listenelementen, die eine Aktivierung im Kanal bewirkten, werden zufällige Aktivierungen im Kontext erzeugt. Diese bewirken ebenfalls, dass die langzeit- und kurzzeit-Gewichte auf die beschriebenen Arten verändert werden. Nach dem letzten Listenelement wird die Repräsentation des Aufgabenendes in der Schicht aktiviert. Diese Aktivierung bewirkt ebenfalls Gewichtsveränderungen durch die beschriebenen Lernregeln.

Anschließend folgt die Wiederholungsphase. Dafür wird erneut die Repräsentation des Aufgabenbeginns in der Schicht aktiviert. Die folgende Aktivierung wird anhand der Gewichtsmatrizen berechnet. Bei diesen berechneten Aktivierungen wird die Genauigkeit der Repräsentation im Kanal mittels der Hamming-Distanz zu den entsprechenden Listenelementen berechnet. Das Listenelement mit der geringsten

Distanz wird als Output erkannt. Mit der neuen Aktivierung der Schicht wird anschließend wiederum das nächste Listenelement berechnet, bis in der Schicht die Repräsentation des Aufgabenendes erkannt wird.

5 Evaluation

Die beschriebenen Eigenschaften und die Funktionsweise des neuronalen Netzes lassen Schlussfolgerungen über die Reaktionsweisen des Modells zu. Diese werde ich in diesem Kapitel zusammenfassen, bevor im folgenden Kapitel konkrete Simulationen durchgänge mit ihren Ergebnissen vorgestellt werden. Die Ergebnisse der Simulationen durchgänge belegen durch praktische Anwendung der Funktionen und des Aufbaus des Modells die theoretischen Vorhersagen.

5.1 Modellvorhersagen

5.1.1 Der „list length“-Effekt

Eine grundlegende Vorhersage des Modells ist, dass die Kapazität für die Speicherung von Listenelementen begrenzt ist. Die Lernregel für die längerfristige Speicherung erzeugt mit zunehmender Listenlänge eine zunehmende Zahl von verstärkten Verbindungen. Da die Trägheit der Gewichte dadurch simuliert wird, dass einmal veränderte Gewichte für die Dauer der Aufgabe nicht mehr überschrieben werden, vermehrt sich die Anzahl der verstärkten Verbindungen mit jedem präsentierten Element. Das bewirkt wiederum, dass in der Wiederholungsphase aus einem aktuellen Zustand diffusere Folgezustände berechnet und somit erhöhte Fehlerzahlen hervorgerufen werden. Die Lernregel für die kurzfristige Speicherung bewirkt, dass in der Präsentationsphase vorherige Veränderungen durch die Interferenzen überschrieben werden. Wenn sich beispielsweise die Repräsentationen im Durchschnitt zu 50% überschneiden wird eine 100%ig Repräsentation eines Folgezustandes in den Kurzzeit-Gewichten nach der Präsentation eines weiteren Elementes auch im Durchschnitt zu 50% überschrieben. Nach einer weiteren Präsentation bleiben durchschnittlich nur noch 25% der gespeicherten Information übrig. Je nach Überschneidung der Repräsentationen bewirken diese Interferenzen, dass nur ein bis zwei Folgezustände mit hinreichender Deutlichkeit zeitgleich gespeichert sein können.

Somit werden die Simulationen des Modells einen „list length“-Effekt aufweisen. Dieser wird mit zunehmendem Anteil der gleichzeitig aktivierten Units stärker ausgeprägt ausfallen, indem nur noch kürzere Listen wiederholt werden können.

Die Größe der Schicht bestimmt, wie viele verschiedene Repräsentationen gespeichert werden können. Eine geringe Schichtgröße bewirkt, dass nur wenige un-

terschiedliche Listenelemente gespeichert werden können. Somit beeinflusst die Größe der Schicht auch die Länge der Listen, die korrekt wiederholt werden können.

5.1.2 Der „primacy“-Effekt

Aus den beschriebenen Lernregeln ergeben sich auch die Hintergründe für einen „primacy“-Effekt und einen „recency“-Effekt.

Die Lernregel für die längerfristigen Gewichte bewirken, dass während der Präsentation der Liste in der Gewichtsmatrix die Zusammenhänge zwischen den einzelnen Listenelementen alle in gleicher Form und Stärke gespeichert werden. Dadurch repräsentiert die Matrix der längerfristigen Gewichte eine Überlagerung aller Zusammenhänge zwischen den aufeinander folgenden Elementen. Wenn sich die Repräsentationen von zwei Listenelementen überschneiden, werden bei der Berechnung des Nachfolgers des einen Listenelementes auch Teile der Repräsentation des Nachfolgers des anderen Listenelementes hinzugefügt.

Die längerfristigen Gewichte erzeugen also während der Wiederholungsphase immer diffusere Repräsentationen, wodurch die Fehlerzahl mit zunehmender Listenposition ebenfalls zunimmt. Somit wird beim Wiederholen des ersten Listenelements noch eine eindeutige Repräsentation berechnet, was die Ursache für das Auftreten eines „primacy“-Effekts ist.

5.1.3 Der „recency“-Effekt

Das Auftreten des „recency“-Effekts wird durch die Lernregeln für die kurzfristigen Gewichte erzeugt. Werden die Zusammenhänge zweier Listenelemente in den kurzfristigen Gewichten gespeichert, überschreiben sie dabei die Gewichte von den Zusammenhängen von Listenelementen, deren Repräsentationen sich überschneiden.

Somit beinhaltet die Gewichtsmatrix eine klare Repräsentation des letzten gespeicherten Zusammenhangs. Die anderen Zusammenhänge wurden, je nach Überschneidung der Repräsentationen der Listenelemente, zu Teilen überschrieben. Die nachfolgenden Listenelemente werden während der Wiederholungsphase aus der Summe der beiden Gewichtsmatrizen berechnet. Zur ansteigenden diffusen Repräsentation des jeweiligen Nachfolgeelementes aus den längerfristigen Gewichten wird dem entsprechend eine zunehmende klare Repräsentation des jeweiligen Nachfolgeelementes aus den Kurzzeit-Gewichten addiert. Damit bewirken die kurzfristigen Gewichte, dass beim Wiederholen der letzten Elemente weniger Fehler gemacht

werden, als bei Elementen aus der Listenmitte. So erklärt die Lernregel für die Kurzzeit-Gewichte das Auftreten eines „recency“-Effekts.

5.1.4 „Errors of omission“

Die verteilte Repräsentation der Listenelemente im Kanal in Verbindung mit der Repräsentation im Kontext erklärt auch das mögliche Auftreten von Auslassungsfehlern. Repräsentationen, die sich überschneiden, führen dazu, dass Gewichte überschrieben werden und dass Teile von mehreren Repräsentationen in einer momentanen Aktivierung enthalten sein können.

Ein Aktivierungsmuster im Kanal stellt die Repräsentation eines Listenelementes dar. Das Aktivierungsmuster im Kontext entspricht der Repräsentation eines Zustandes des Netzes, der von den einzelnen Listenelementen unabhängig ist. Da dieses Aktivierungsmuster keine Informationen über das aktuelle Listenelement beinhaltet trägt sie lediglich zur Berechnung des Folgezustandes der gesamten Schicht bei. Das heißt, dass eine Repräsentation, die, im Vergleich zur präsentierten Liste, nur Fehler in der Aktivierung des Kontexts besitzt, eine fehlerfreie Repräsentation des Listenelements aufweist. Die Fehler im Kontext wirken sich allerdings bei der Berechnung des Folgeschrittes auch auf den Kanal aus, so dass im Folgeschritt eine fehlerhafte Repräsentation des Listenelementes auftreten kann. Die Aktivierung des Kanals repräsentiert ein einzelnes Listenelement, trägt aber auch zur Berechnung des Folgezustandes der gesamten Schicht bei.

In der momentanen Aktivierung des Kanals ist also das Listenelement kodiert. In den Gewichten der gesamten Schicht werden die nachfolgenden Aktivierungen und somit die Reihenfolge der Liste kodiert. Fehler und ungenaue Repräsentationen können dazu führen, dass das Listenelement mit einem anderen verwechselt wird, oder auch, dass die Listenposition verwechselt wird. „Errors of omission“ treten auf, wenn die momentane Repräsentation der Listenposition in der Wiederholungsphase mit einer folgenden Listenposition verwechselt wird oder aber gar nicht mehr einer Position der Liste zugeordnet werden kann. Im ersten Fall wird dabei an der folgenden Listenposition mit der Wiederholung der Liste fortgefahren. Im zweiten Fall kann die restliche Liste jedoch nicht mehr wiederholt werden und die Ausgabe endet nach dem aktuellen Element.

5.1.5 „Errors of commission“

Da die Listenelemente nur im Kanal repräsentiert werden treten „errors of commission“ nur dann auf, wenn die aktuelle Repräsentation im Kanal einem Element ähnlicher ist, als das eigentlich folgende Element der Liste. Dann wird das ähnlichere Element fälschlicherweise ausgegeben. Die momentane Repräsentation im Kontext kann aber trotzdem der Aktivierung entsprechen, die zusammen mit dem Listenelement in der Präsentationsphase stattfand. Dann trägt das Aktivierungsmuster im Kontext anschließend dazu bei, dass nach dem Fehler trotzdem mit dem korrekten Listenelement fortgefahren wird. Somit wird an einer Stelle der Liste ein falsches Element wiederholt, ohne dass die Elemente an den restlichen Positionen der Liste betroffen sind. Wenn nun die Listenelemente reichhaltiger kodiert werden sollten, beispielsweise durch eine bildhaftere Vorstellung, wird dies zu einem größeren Kanal für die Aufgabe führen und die Anzahl der „errors of commission“ reduzieren.

5.2 Die Simulationen

Bei allen Simulationen betrug die Größe der Schicht 200 Neuronen und die Repräsentation eines Listenelementes waren Aktivierungsmuster bei denen 25% der Schicht aktiviert waren. Eine Menge dieser Aktivierungsmuster wurde vor jedem Durchlauf zufällig generiert. Diese Menge entspricht den möglichen Listenelementen und betrug bei allen Simulationen die Größe 26 analog zu den 26 Buchstaben des Alphabets.

Eine größere Neuronenanzahl in der Schicht bewirkt, dass das Netz bei der Wiederholung längerer Sequenzen weniger Fehler macht als mit einer geringeren Neuronenanzahl. Dem entgegen wirkt der Anteil der Aktivierung der Repräsentationen. Ein höherer Anteil an Aktivierung bewirkt höhere Fehlerzahlen bei gleicher Listenlänge. Jede Simulation wurde mit den jeweiligen Parametereinstellungen 1000-mal wiederholt.

Insgesamt habe ich vier verschiedene Simulationen durchgeführt um die verschiedenen Effekte und deren Hintergründe aufzuzeigen.

5.2.1 Simulation 1

Als erstes variierte ich die Listenlänge zwischen Listen mit nur einem einzigen Element und Listen mit 11 Elementen. Der Anteil des Kanals betrug 50 %. In Tabelle 1 sind die Ergebnisse der Simulation zusammengefasst. Mit zunehmender Listenlänge nimmt der Anteil der korrekt wiederholten Listen stark ab. Dies entspricht dem „list length“-Effekt. Abbildung 10 zeigt die Werte aus Tabelle 1 als graphische Darstellung. Es ist eindeutig ein sigmoidaler Verlauf der Kurve zu erkennen. Diese ähnelt den empirischen Ergebnissen von Crannell & Parrish (1957) (vgl. Kapitel 2.2.1). Diese haben mit ihren Untersuchungen von „serial-recall“-Aufgaben mit unterschiedlichen Listenelementen ebenfalls sigmoidale Kurvenverläufe für den „list length“-Effekt entdeckt, die dem Simulationsergebnis sehr ähnlich sind.

Listenlänge	Anteil der korrekt wiederholten Listen
1	100%
2	100%
3	100%
4	99,2%
5	99%
6	93,2%
7	77,3%
8	41,5%
9	9,7%
10	0,7%
11	0%

Tabelle 1: Anteile der korrekt wiederholten Listen unterschiedlicher Länge

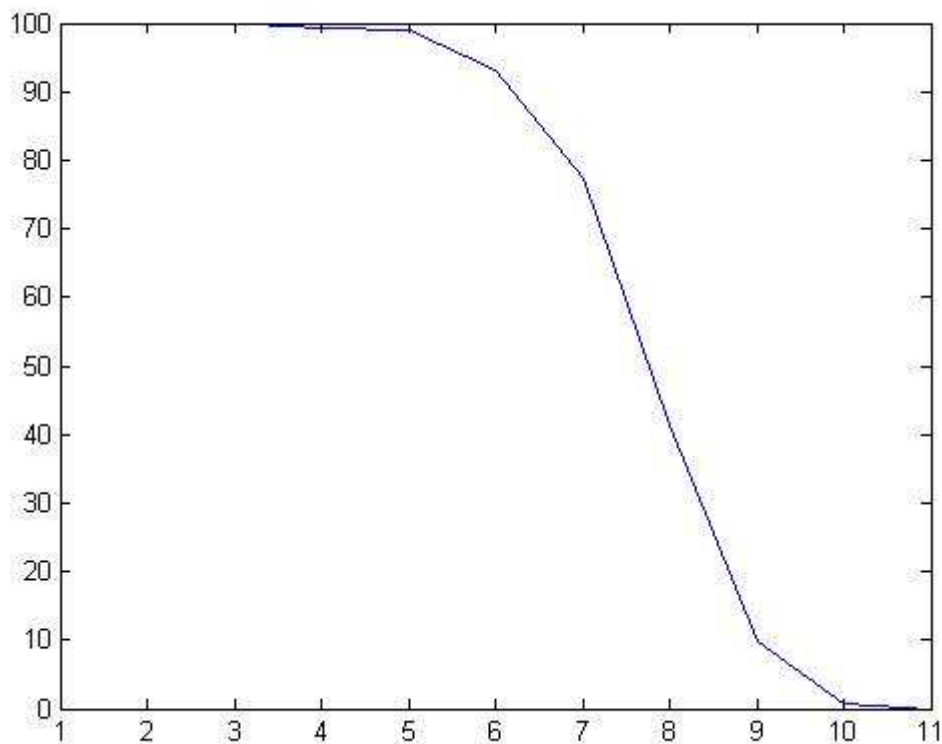


Abbildung 10: Anteile der korrekt wiederholten Listen unterschiedlicher Länge

5.2.2 Simulation 2

Als nächstes wählte ich eine Liste der Länge 7, da bei dieser Listenlänge noch ca. 77% korrekt erkannt werden, aber doch schon eine deutliche Anzahl an Fehlern gemacht werden. Der Kanal des Netzes betrug, wie bei der vorherigen Simulation, 50 % der Schicht. Die Abbildung 11 zeigt die durchschnittliche Ähnlichkeit auf Basis der Hamming-Distanz der einzelnen Aktivierungsmuster zu den Repräsentationen der entsprechenden Listenelemente während der Wiederholungsphase. Sowohl ein „primacy“-Effekt als auch ein „recency“-Effekt sind deutlich ausgeprägt. Der „recency“-Effekt tritt dabei etwas schwächer auf als der „primacy“-Effekt. Diese Form entspricht den empirischen Untersuchungsergebnissen (z.B. Baddeley, 1999, Cowan, 2005, Botvinick & Plaut, 2006). Bei anderen Listenlängen treten gleiche Kurvenverläufe auf.

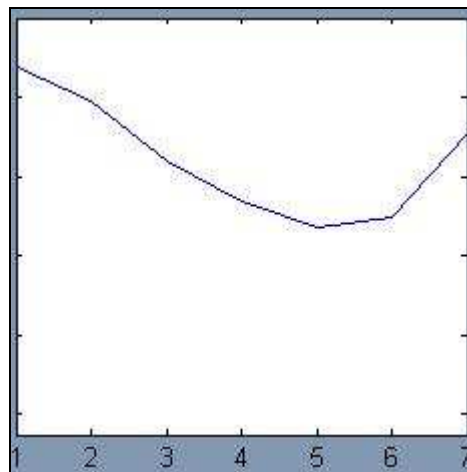


Abbildung 11: Durchschnittliche Ähnlichkeit der einzelnen Aktivierungsmustern zu den Repräsentationen der Listenelementen

5.2.3 Simulation 3

In dieser Simulation habe ich den Einfluss der Kurzzeit-Gewichte ausgeschaltet, indem ich bei der Berechnung der Liste ausschließlich die Gewichtsmatrix für die Langzeit-Gewichte verwendet habe. In Abbildung 12 wird deutlich, dass der „recency“-Effekt unter diesen Bedingungen nicht mehr auftritt. Somit zeigt sich, dass die Kurzzeit-Gewichte für den „recency“-Effekt verantwortlich sind. Der „primacy“-Effekt zeigt sich jedoch in unveränderter Form, womit die Modellvorhersage, dass die Langzeit-Gewichte für dieses Phänomen verantwortlich sind, unterstrichen wird.

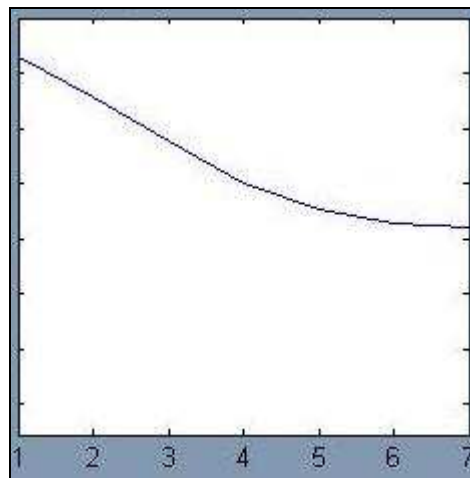


Abbildung 12: Durchschnittliche Ähnlichkeit ohne Kurzzeit-Gewichte

5.2.4 Simulation 4

Bei allen Simulationen traten sowohl „errors of omission“ als auch „errors of commission“ auf. In der letzten Simulation variierte ich den Anteil des Kanals an der Gesamtschicht, um die Folgen für das Verhältnis der beiden Fehlerarten zueinander zu untersuchen. In den Abbildungen 13 und 14 sind die Fehlerverhältnisse für Kanalanteile von 50 %, 40 %, 30 % und 20 % dargestellt. Dabei treten bei allen Durchläufen deutlich mehr „errors of omission“ als „errors of commission“ auf. Dieses ähnelt den Aussagen der Abbildung 2 von Schwartz et al. (1991) bei großen Fehlerzahlen. Man kann aber auch sehen, dass mit abnehmender Kanalgröße der Anteil der „errors of commission“ zunimmt, wie es in den Modelvorhersagen prognostiziert wurde. Dies ähnelt den Aussagen, dass eine umfassendere Vorstellung dazu führt, dass sowohl längere Listen gespeichert werden können, als auch weniger Verwechslungen der Elemente solcher Listen auftreten (Baddeley, 1999)

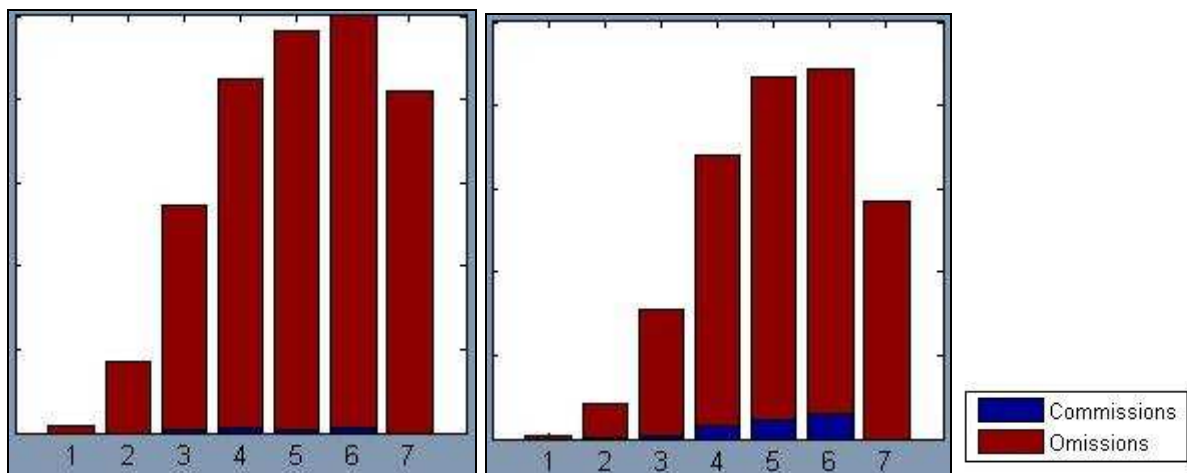


Abbildung 13: Fehlerverhältnis bei Kanalgrößen von 50 % (links) und 40 % (rechts) der Schicht

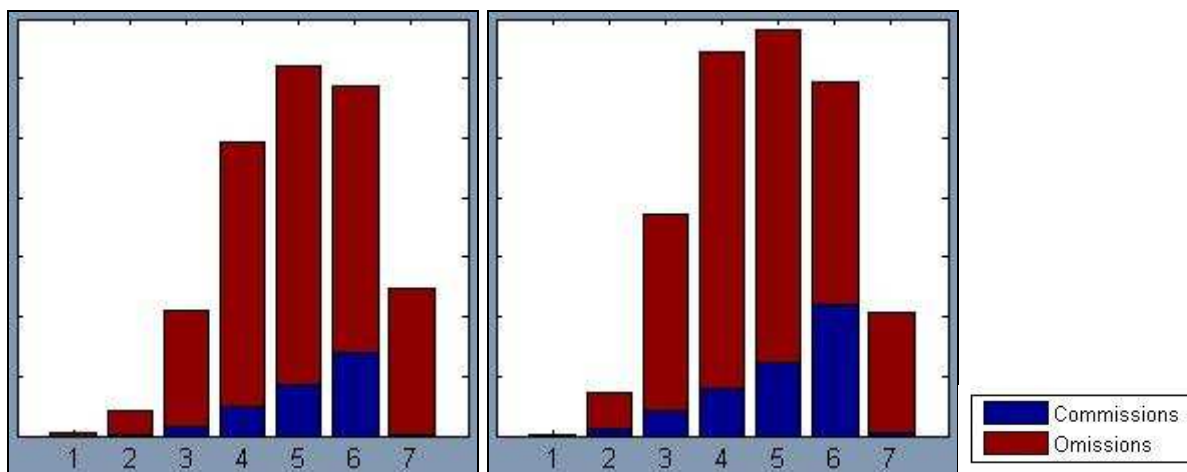


Abbildung 14: Fehlerverhältnis bei Kanalgrößen von 30 % (links) und 20 % (rechts) der Schicht

6 Zusammenfassung und Ausblick

In der vorliegenden Diplomarbeit habe ich gezeigt, dass aktuelle Modelle Probleme bei der Erklärung der Effekte haben, die bei der Aufgabe des „immediate serial recall“ auftreten. Dabei habe ich mich bei den auftretenden Effekten auf den „list length“-Effekt, den „primacy“-Effekt und den „recency“-Effekt sowie auf die „errors of omission“ und die „errors of commission“ konzentriert (vgl. Kapitel 2.2).

In den theoretischen Modellen von Baddeley und Logie (vgl. Kapitel 3.1), Cowan (vgl. Kapitel 3.2) sowie von Ericsson und Kintsch (vgl. Kapitel 3.3) können die angesprochenen Effekte nur durch vage Beschreibungen und Lücken, die Spielraum für mögliche Mechanismen lassen, erklärt werden. Das Modell von Botwinick und Plaut (vgl. Kapitel 3.4) kann zwar als einziges beschriebenes Modell alle Effekte erklären, bedarf hierfür allerdings einer komplexen Struktur mit einer großen Anzahl an Funktionen. Diese widersprechen außerdem den Erkenntnissen über neurobiologische Vorgänge bei der Informationsverarbeitung im Gehirn.

Das von mir eingeführte Modell (vgl. Kapitel 4) zeichnet sich dadurch aus, dass es mit einer ausgesprochen einfachen Struktur mit nur einer einzigen Verarbeitungsschicht und lediglich zwei unterschiedlichen Lernfunktionen alle beschriebenen Effekte erklären kann. Dabei ist das neuronale Netz, im Vergleich zum Modell von Botwinick und Plaut, nicht auf eine vorherige Trainingsphase angewiesen. Außerdem wurden alle Eigenschaften anhand neurobiologischer Erkenntnisse hergeleitet.

Somit kann das Modell, welches im Rahmen dieser Arbeit entwickelt wurde, im Gegensatz zu den vorgestellten theoretischen Modellen, konkrete Ursachen für die beschriebenen Effekte nennen, und diese mit geringerem Aufwand erklären, als das Modell von Botwinick und Plaut (2006).

Als nächster Schritt würde die Ausweitung des neuen Modells auf weitere Effekte bei der Aufgabe des „immediate serial recall“ folgen. Anschließend könnte die Umsetzung auf weitere Aufgaben folgen. Dabei müssten alle Eigenschaften hinzugefügt werden, die entweder zur Bearbeitung der „immediate serial recall“-Aufgabe nicht benötigt wurden oder zur Vereinfachung ausgelassen wurden. Beispielsweise müsste dann der Einfluss der quasi-permanent gespeicherten Informationen modelliert werden. Eine weitere Möglichkeit der Erweiterung des Modells wäre, es mit Sensoren auszustatten, die eine innere Repräsentation der Umwelt erzeugen.

7 Literaturverzeichnis

- Baddeley, A. D. (1986). *Working Memory*.
- Baddeley, A. D. (1996). Exploring the central executive, *The Quarterly Journal of Experimental Psychology*, 49 A (1), S. 5-28.
- Baddeley, A. D. (1999). *Essentials of Human Memory*. Psychology Press
- Baddeley, A. D. (2000). The episodic buffer: A new component of working memory? *Trends in Cognitive Sciences*, 4 (11), S. 417-423.
- Baddeley, A. D. (2002). Is working memory still working? *European Psychologist*, 7 (2), S. 85-97.
- Baddeley, A. D., Hitch, G. J. (1974). Working memory, In G.A. Bower (Ed.), *The psychology of learning and motivation: advances in research and theory*, 8, S. 47-89. New York: Academic Press.
- Baddeley, A. D., Hitch, G. J. (1993). The recency effect: Implicit learning with explicit retrieval? *Memory & Cognition*, 21 (2), S. 146-155.
- Baddeley, A. D., Logie, R. H. (1999). Working memory: The multiple-component model. In A. Miyake, P. Shah (Hrsg.), *Models of Working Memory*, S. 28-61. Cambridge University Press.
- Baddeley, A. D., & Wilson, B., (1993). A case of word deafness with preserved span: Implications for the structure and function of working memory. *Cortex*, 29, S. 741-748.
- Bisiach, E. (1993). Mental representation in unilateral neglect and related disorders: The twentieth Bartlett memorial lecture. *Quarterly Journal of Experimental Psychology*, 46A, S. 435-461.
- Black, I. B. (1993). *Symbole, Synapsen und Systeme*. Heidelberg, Berlin, Oxford: Spektrum, Akad. Verl.
- Botvinick, M. M. (2005). Effects of domain-specific knowledge on memory for serial order, *Cognition*, 97, S. 135-151.
- Botvinick, M. M., Plaut, D. C. (2004). Doing without schema hierarchies: A recurrent connectionist approach to normal and impaired routine sequential action, *Psychological Review*, 111 (2), S. 395-429.
- Botvinick, M. M., Plaut, D. C. (2006). Short-term memory for serial order: A recurrent neural network model, *Psychological Review*, 113 (2), S. 201-233.
- Broadbent, D. E., (1958). *Perception and communication*. London: Pergamon Press.

- Brooks, L. R. (1967). The suppression of visualisation by reading. *Quarterly Journal of Experimental Psychology*, 19, S.289-299.
- Caplan, D., Rochon, E., & Waters, G. S.(1992). Articulatory and phonological determinants of word length effects in span tasks. *Quarterly Journal of Experimental Psychology*, 45A, S. 177-192.
- Chase, W. G., Ericsson, K. A. (1982). Skill and working memory. In G. H. Bower (Ed.), *The psychology of learning and motivation*, 16, S. 1-58. New York: Academic Press.
- Conrad, R. (1959). Errors of immediate memory, *British Journal of Psychology*, 51, S. 45-48.
- Cornoldi, C., Cortesi, A., & Preti, D. (1991). Individual differences in the capacity limitations of visuospatial short-term memory: Research on sighted and totally congenitally blind people. *Memory & Cognition*, 19, S. 459-468.
- Cowan, N. (1992). Verbal memory span and the timing of spoken recall, *Journal of Memory and Language*, 31, S. 668-684.
- Cowan, N. (1995). *Attention and Memory: An Integrated Framework*. New York: Oxford University Press.
- Cowan, N. (1999). An embedded-process model of working memory. In A. Miyake, P. Shah (Hrsg.), *Models of Working Memory*, S. 62-101. Cambridge University Press.
- Cowan, N. (2001). The magical number 4 in short-term memory: A reconsideration of mental storage capacity, *behavioral and brain sciences*, 24 (1), S. 87 -185.
- Cowan, N. (2005). *Working Memory Capacity*. New York: Psychology Press.
- Cowan, N., Sauls, J. S., Elliott, E. M., Moreno, M. V. (2002). Deconfounding serial – recall, *Journal of Memory and Language*, 46, S. 153-177.
- Cowan, N., Wood, N. L., Nugent, L. D., Treisman, M. (1997). There are two word-length effects in verbal short-term memory: Opposed effects of duration and complexity, *Psychological Science*, 8 (4), S. 290-295.
- Crannell, C. W., & Parrish, J. M. (1957). A comparison of immediate memory span for digits, letters, and words. *Journal of Psychology*, 51, S. 319-327.
- Daneman, M., & Carpenter, P. A. (1980). Individual differences in working memory and reading. *Journal of Verbal Learning and Verbal Behavior*, 19, S. 450-466.
- Davies, D. R., & Parasuaman, R. (1982). *The psychology of vigilance*. New York: Academic Press.

- de Groot, A. (1946/1978). *Thought and choice in chess*. The Hague: Mouton.
- Duff, S. C., & Logie, R. H. (2001). Processing and storage in working memory span, *The quarterly journal of experimental psychology*, 54A (1), S. 31-48.
- Ebbinghaus, H., (1885/1992). *Über das Gedächtnis: Untersuchungen zur experimentellen Psychologie*. Darmstadt: Wissenschaftliche Buchgesellschaft.
- Ericsson, K. A., Delaney P. F. (1999). Long-term working memory as an alternative to capacity models of working memory in everyday skilled performance. In A. Miyake, P. Shah (Hrsg.), *Models of Working Memory,, Models of Working Memory*, S. 257-297. Cambridge University Press.
- Ericsson, K. A., Kintsch, W. (1995). Long-term working memory, *Psychological Review* , 102 (2), S. 211-245.
- Garner, B., (2003). A novel approach to training neurons with biological plausibility, *Neurocomputing*, 52-54, S. 265-270.
- Gathercole, S., Baddeley, A. D., (1993). *Working memory and language*. Hove, UK: Erlbaum.
- Hebb, D. O. (1949). *The organisation of behavior*. New York: John Wiley & Sons.
- Jahnke, J. C. (1963). Serial position effects in immediate serial recall, *Journal of Verbal Learning and Verbal Behaviour*, 2, S.284-287.
- Kieras, D. E., Meyer, D. E. (1994). *The EPIC achitecture for modelling human information-processing and performance: A brief introduction*. Ann Arbor: Univeristy of Michigan.
- Lashley, K. S. (1951). The problem of serial order in behavior. In L. A. Jeffress (Ed.), *Cerebral mechanisms in behavior: The Hixon symposium*, (S. 112-136). New York: Wiley.
- Logie, R. H. (1995). *Visio-Spatial Working Memory*.
- Logie, R. H., Baddeley, A. D., Mane, A., Donchin, E. & Sheptak, R. (1989). Working memory and the analysis of a complex skill by secondary task methodology. *Acta Psychologica*, 71, S. 53-87.
- Logie, R. H., & Marchetti, C. (1991). Visuo-spatial working memory: Visual, spatial or central executive? In R. H. Logie & M. Denis (Ed.), *Mental images in human cognition* (S. 105-115). Amsterdam: Elsevier.
- Miller, G. A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information, *The Psychological Review*, 63, S. 81-97.

- Miller, G. A., & Selfridge, J. A. (1951). Verbal context and the recall of meaningful material. *American Journal of Psychology*, 63, S. 176-185.
- Miyake, A., Shah, P. (1999) *Models of Working Memory*. Cambridge: University Press.
- Newell, A. (1990). *Unified theories of cognition*. Cambridge, MA: Harvard University Press.
- Nipher, F. E. (1876). On the distribution of errors in numbers written from memory. *The Transactions of the Academy of Science of Saint Louis*, 3, S. 210-211.
- Pinel, J. P. J., Pauli, P.(2007). *Biopsychologie*. München: Pearson Education Deutschland GmbH.
- Purves, D., Augustine, G. J., Fitzpatrick, D., Hall, W. C., LaMantia, A.-S., McNamary, J. O., Williams, S. M.(2004). *Neuroscience*. Sunderland: Sinauer Associates.
- Rumelhart, D. E. (1986). *Parallel distributed processing: Explorations in the microstructure of cognition*. Cambridge: MIT Press.
- Saariluoma, P. (1989). Chess players' recall of auditorily presented chess positions. *European Journal of Cognitive Psychology*, 1, S. 309-330.
- Schwarz, M. F., Reed, E. S., Montgomery, M. W., Palmer, C., & Mayer, N. H. (1991). The quantitative description of action disorganization after brain damage: A case study. *Cognitive Neuropsychology*, 8, S. 381-414.
- Shallice, T., Warrington, E. K., (1970). Independent functioning of verbal memory stores: A neuropsychological study. *Quarterly Journal of Experimental Psychology*, 41, S. 261-273.
- Sweatt, J. D.(2003). *Mechanisms of Memory*, San Diego, CA: Elsevier.
- Smith, E. E., Jonides, J. (1997). Working memory: A view from neuroimaging. *Cognitive Psychology*, 33, S. 5-42.
- Solokov, E. N. (1963). *Perception and the conditioned reflex*. New York: Pergamon Press.
- Teasdale, J. D., Dritschel, B. H., Taylor, M. J., Proctor, L., Lloyd, C. A., Nimmo-Smith, I., & Baddeley, A. D. (1995). Stimulus-independent thought depends on central executive resources. *Memory & Cognition* 23, S. 551-559.
- Vallar, G., & Baddeley, A. D. (1995). Fractionation of working memory: Neuropsychological evidence for a phonological short store. *Journal of Verbal Learning and Verbal Behavior*, 23, S. 151-161.

Williams, R. J., Zipser, D. (1995). Gradient-based learning algorithms for recurrent neural networks and their computational complexity. In Y. Chauvin & D. E. Rumelhart (Ed.), *Backpropagation: Theory, architectures, and applications* (S. 433-486). Hillsdale, NJ: Erlbaum.

8 Anhang

8.1 Quellcode

8.1.1 Start.m

```
function varargout = Start(varargin)
% START M-file for Start.fig
%     START, by itself, creates a new START or raises the existing
%     singleton*.
%
%     H = START returns the handle to a new START or the handle to
%     the existing singleton*.
%
%     START('CALLBACK',hObject,eventData,handles,...) calls the local
%     function named CALLBACK in START.M with the given input arguments.
%
%     START('Property','Value',...) creates a new START or raises the
%     existing singleton*. Starting from the left, property value pairs
are
%     applied to the GUI before Start_OpeningFcn gets called. An
%     unrecognized property name or invalid value makes property applica-
tion
%     stop. All inputs are passed to Start_OpeningFcn via varargin.
%
%     *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only one
%     instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help Start

% Last Modified by GUIDE v2.5 28-Feb-2008 11:08:33

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @Start_OpeningFcn, ...
                  'gui_OutputFcn',  @Start_OutputFcn, ...
                  'gui_LayoutFcn',  [] , ...
                  'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before Start is made visible.
function Start_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
```

```

% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to Start (see VARARGIN)

% Choose default command line output for Start
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes Start wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = Start_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

function lsValue_Callback(hObject, eventdata, handles)
% hObject    handle to lsValue (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of lsValue as text
%         str2double(get(hObject,'String')) returns contents of lsValue as a
double

x= str2double(get (hObject, 'String' ) );
x= round(x);
set (hObject, 'String' ,num2str(x)) ;

% --- Executes during object creation, after setting all properties.
function lsValue_CreateFcn(hObject, eventdata, handles)
% hObject    handle to lsValue (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function setSValue_Callback(hObject, eventdata, handles)
% hObject    handle to setSValue (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

```

```

% Hints: get(hObject,'String') returns contents of setSValue as text
%         str2double(get(hObject,'String')) returns contents of setSValue as
a double

x=str2double( get ( hObject, 'String' ) );
x=round(x);
if x< str2double(get ( handles.seqValue, 'String' ))
    x=str2double(get ( handles.seqValue, 'String' ));
end
set ( hObject, 'String', num2str(x) );

% --- Executes during object creation, after setting all properties.
function setSValue_CreateFcn(hObject, eventdata, handles)
% hObject    handle to setSValue (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function seqValue_Callback(hObject, eventdata, handles)
% hObject    handle to seqValue (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of seqValue as text
%         str2double(get(hObject,'String')) returns contents of seqValue as
a double

x=str2double( get ( hObject, 'String' ) );
x=round(x);
if x> str2double(get ( handles.setSValue, 'String' ))
    set ( handles.setSValue, 'String', num2str(x) );

    x=str2double(get ( handles.setSValue, 'String' ));
end
set ( hObject, 'String', num2str(x) );

% --- Executes during object creation, after setting all properties.
function seqValue_CreateFcn(hObject, eventdata, handles)
% hObject    handle to seqValue (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

```

```

function nValue_Callback(hObject, eventdata, handles)
% hObject      handle to nValue (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of nValue as text
%         str2double(get(hObject,'String')) returns contents of nValue as a
double

x= str2double(get (hObject, 'String' ) );
x= round(x);
set (hObject, 'String' ,num2str(x)) ;

% --- Executes during object creation, after setting all properties.
function nValue_CreateFcn(hObject, eventdata, handles)
% hObject      handle to nValue (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in startButton.
function startButton_Callback(hObject, eventdata, handles)
% hObject      handle to startButton (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)

seq=str2double(get(handles.seqValue, 'String'));
ls=str2double(get(handles.lsValue, 'String'));
ar=str2double(get(handles.arValue, 'String'));
cr=str2double(get(handles.crValue, 'String'));
setS=str2double(get(handles.setSValue, 'String'));
n=str2double(get(handles.nValue, 'String'));

e=loop(seq,ls,ar,cr,setS,n);

axes(handles.axes1);
plot(-e.errors(1:size(e.errors,2)-1))
xlabel('Listenelement')
ylabel('Genauigkeit')
title('Genauigkeiten der Listenelemente')
axis([1 max(size(e.errors,2)-1,2) min(min(-e.errors(1:size(e.errors,2)-
1))*2,-0.1) 0]);

axes(handles.axes2);
bar([e.Commissions;e.Omissions'],'stack');
%bar(e.Commissions,'stack');
xlabel('Listenelement')
ylabel('Fehleranzahl')
title('Fehleraufteilung')
legend('Commissions','Omissions');

set ( handles.corText, 'String', e.correct );

```

```

% --- Executes on button press in exitButton.
function exitButton_Callback(hObject, eventdata, handles)
% hObject    handle to exitButton (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
delete(handles.figure1);
%exit;

% --- Executes on slider movement.
function arSlider_Callback(hObject, eventdata, handles)
% hObject    handle to arSlider (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'Value') returns position of slider
%        get(hObject,'Min') and get(hObject,'Max') to determine range of
slider

x= get ( hObject, 'Value' ) ;
x= round(x*20)/20;
set ( hObject, 'Value' ,x) ;
set ( handles.arValue, 'String', num2str ( get ( hObject, 'Value' ) ) );

% --- Executes during object creation, after setting all properties.
function arSlider_CreateFcn(hObject, eventdata, handles)
% hObject    handle to arSlider (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function arValue_Callback(hObject, eventdata, handles)
% hObject    handle to arValue (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of arValue as text
%        str2double(get(hObject,'String')) returns contents of arValue as a
double

x=str2double( get ( hObject, 'String' ) );
if x > get ( handles.arSlider, 'Max' )
    x=get ( handles.arSlider, 'Max' );
else
    if x < get ( handles.arSlider, 'Min' )
        x=get ( handles.arSlider, 'Min' );
    end
end
set ( handles.arSlider, 'Value', x );
set ( hObject, 'String', num2str(x) );

% --- Executes during object creation, after setting all properties.

```

```

function arValue_CreateFcn(hObject, eventdata, handles)
% hObject    handle to arValue (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on slider movement.
function crSlider_Callback(hObject, eventdata, handles)
% hObject    handle to crSlider (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'Value') returns position of slider
%         get(hObject,'Min') and get(hObject,'Max') to determine range of
slider

x= get ( hObject, 'Value' ) ;
x= round(x*20)/20;
set ( hObject, 'Value' ,x) ;
set ( handles.crValue, 'String', num2str ( get ( hObject, 'Value' ) ) );

% --- Executes during object creation, after setting all properties.
function crSlider_CreateFcn(hObject, eventdata, handles)
% hObject    handle to crSlider (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function crValue_Callback(hObject, eventdata, handles)
% hObject    handle to crValue (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of crValue as text
%         str2double(get(hObject,'String')) returns contents of crValue as a
double
x=str2double( get ( hObject, 'String' ) );
if x > get ( handles.crSlider, 'Max' )
    x=get ( handles.crSlider, 'Max' );
else
    if x < get ( handles.crSlider, 'Min' )
        x=get ( handles.crSlider, 'Min' );
    end
end
set ( handles.crSlider, 'Value', x );
set ( hObject, 'String', num2str(x) );

```

```
% --- Executes during object creation, after setting all properties.
function crValue_CreateFcn(hObject, eventdata, handles)
% hObject    handle to crValue (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

8.1.2 loop.m

```
function e=loop(seq,ls,ar,cr,setS,n)
%Durchlaeuft n Simulationen von jeweils neu generierten Netzen und
%berechnet die unterschiedlichen Fehlerzahlen
%
%Parameter: seq=Laenge der Sequenz
%           ls= Groesse der Schicht
%           ar= Anteil der Aktivierung
%           cr= Anteil des Kanals an der Schicht
%           setS= Mengengroesse fuer die moeglichen Listenelemente
%           n= Anzahl der Durchlaeufer

errsum=zeros(1,seq+1);
commission=zeros(1,seq);
omission=zeros(1,seq);
cor=0;
corLen=0;

for i=1:n
    r=auto(seq,ls,ar,cr,setS);

    if size(r.outputs,2)>0
        k=1;
        for j=1:size(r.inputs,2)
            if r.outputel(k)~=j
                if size(r.outputel,2)>k
                    if r.outputel(k)>j && r.outputel(k+1)~= j+1 %&&
all(r.outputel(k+1:size(r.outputel,2))>=r.outputel(k))
                        omission(j)=omission(j)+1;
                    else if j+1==r.outputel(k+1)
                        commission(j)=commission(j)+1;
                        k=k+1;
                    end
                end
            else
                omission(j)=omission(j)+1;
                %Werden am meisten gemacht
            end
        else if size(r.outputel,2)>k
            k=k+1;
        else
            if j~=size(r.inputs,2)
                omission(j)=omission(j)+1;
                %Werden am Anfang gemacht
            end
        end
    end
end
end

if size(r.outputs,2)==seq;
    errsum=errsum+r.error;
    corLen=corLen+1;
    if isequal(1:size(r.outputs,2),r.outputel)
        cor=cor+1;
    end
end
end
end
```

```
e.Commissions=commission;  
e.Omissions=omission;  
e.errors=errsum/corLen;  
e.correct=cor;
```

8.1.3 auto.m

```
function r=auto(sl,ls,ar,cr,setS)
%automatisches Simulieren des Netzes mit zufaelligen Eingaben
%
%Parameter:      sl(SequenceLength)
%                ls(LayerSize)
%                ar(ActivationRatio)
%                cr(ChannelRatio)
%                setS(SetSize)
%
%Returns:        r(Result)

global parameters;
parameters.additionalsteps=4;           %erlaubte Schritte nach der Sequenz-
länge

global result

result.parameters=parameters;
result.inputs=[];
result.outputs=[];
result.error=[];
result.set=[];
result.outputel=[];
result.ls=ls;
result.cs=round(ls*cr);
result.act=round(ls*ar); %Aktivierung der Schicht
result.cact=round(result.cs*ar); %Aktivierung des Channels
result.kact=result.act-result.cact; %Aktivierung des Kontext
result.weights=cell(3,sl);

result.startvec= [zeros(result.cs,1);randvec(ls-
result.cs,(result.kact)*2)];
result.endvec= [zeros(result.cs,1);randvec(ls-result.cs,(result.kact)*2)];

%Sicherstellen, dass startvector ~= endvector
while isequal(result.startvec,result.endvec)
    result.startvec= [zeros(result.cs,1);randvec(ls-
result.cs,(result.kact)*2)];
    result.endvec= [zeros(result.cs,1);randvec(ls-
result.cs,(result.kact)*2)];
end

network=dwnet(result.cs,ls);

%Lernen Start
network.layer.values=result.startvec;

result.set=randvec(result.cs,result.cact);
while size(result.set,2)<setS
    el= randvec(result.cs,result.cact);
    if ~ismember(el(1:result.cs,1)',result.set(1:result.cs,:))', 'rows')
        result.set=[result.set el];
    end
end
end
```

```

network.layer.values=result.startvec;
for i=1:s1
    v=[result.set(:,i);randvec(ls-result.cs,result.kact)];

    result.inputs=[result.inputs v];
    [network,out]=calc(network,v);
    result.weights{1,i}=network.weights1;
    result.weights{2,i}=network.weights2;
end
[network,out]=calc(network,result.endvec);

%Erinnerung Start
step=0;
network.layer.values=result.startvec;
while (step ~= s1+parameters.additionalsteps) &&
~isequal(network.layer.values,result.endvec)
    step=step+1;
    [network,out]=calc(network);
    if ~isequal(network.layer.values,result.endvec)
        result.outputs=[result.outputs out];
    end
end
r=result;

```

8.1.4 calc.m

```
function [net, outp]=calc(net, inp)
% Berechnet den Output des Netzes fuer den Input inp

global result;

if nargin == 2
    input=inp;

    x=recall(net);
    var=inp-x;

    net=learn(net, input, var);
    net.layer.values=input;
else
    zero=zeros(net.layer.size,1);
    x=recall(net);

    [x,c]=recognize(x);
    result.error=[result.error c];

    net=learn(net, zero, zero); %Kann waehrend des Erinnerns eingeschaltet
    werden
    net.layer.values=x;
end

outp=net.layer.values;
```

8.1.5 dwnet.m

```
function net=dwnet(cs,ls)
% Erstellt neues Netzwerk mit folgenden Parametern:
%
%   cs(ChannelSize)=Anzahl der Neuronen des Channels
%   ls(LayerSize)=Anzahl der Neuronen

net.cs=cs;
net.layer.size=ls;
net.layer.values=zeros(net.layer.size,1);

net.weights1=zeros(net.layer.size,net.layer.size); %kurzzeit-Gewichte
net.weights2=zeros(net.layer.size,net.layer.size); %langzeit-Gewichte
```

8.1.6 learn.m

```
function net=dwnet(cs,ls)
% Erstellt neues Netzwerk mit folgenden Parametern:
%
%   cs(ChannelSize)=Anzahl der Neuronen des Channels
%   ls(LayerSize)=Anzahl der Neuronen

net.cs=cs;
net.layer.size=ls;
net.layer.values=zeros(net.layer.size,1);

net.weights1=zeros(net.layer.size,net.layer.size); %kurzzeit-Gewichte
net.weights2=zeros(net.layer.size,net.layer.size); %langzeit-Gewichte
```

8.1.7 recall.m

```
function v=recall(net)
% Erinnert die naechste Aktivierung des Netzes

global result;

longterm=round(net.weights2)*net.layer.values;
shorttterm=net.weights1*net.layer.values;

v=shorttterm+longterm; %Moeglichkeit einzelne Gewichte auszuschliessen
%v=shorttterm; %Moeglichkeit einzelne Gewichte auszuschliessen
%v=longterm; %Moeglichkeit einzelne Gewichte auszuschliessen

s=sum(max(v,0))/result.ls;      %Schwellenwert = durchschnittliche erregende
Aktivierung

for i=1:size(v,1)
    if v(i)<= s
        v(i)=0;
    else
        v(i)=1;
    end
end
end
```

8.1.8 recognize.m

```
function [v,c]=recognize(v)
% Erkennt aehnlichstes Element aus dem Set
%
%Parameter v(Value)=Aktuelle Aktivierung
%
%Returns c=Abweichung zwischen v und erkanntem Element

global result;

if max(v)<=0
    v=result.endvec;
    c=0;
else
    sim=[];
    value=v;

    for i=1:size(result.set,2)
        sim=[sim sum(abs(value(1:result.cs,1)-result.set(:,i)))];
    end
    if min(sim)>=min(sum(abs(value(result.cs+1:result.ls,1)-
result.endvec(result.cs+1:result.ls,1))))
        c=min(sim,[],2);
        v=result.endvec;
    else
        c=min(sim,[],2);
        x=[];
        for j=1:size(sim,2) %Weitere Elemente mit gleicher Aehnlichkeit
finden
            if sim(1,j)==c
                x=[x j];
            end
        end
        p=randperm(size(x,2)); %Aus Elementen gleicher Aehnlichkeit zufael-
liges auswaehlen
        result.outputel=[result.outputel x(1,p(1,1))];
    end
end
end
```

8.1.9 randvec.m

```
function r=randvec(size,n)

r=zeros(size,1);
x=randperm(size);
for i=1:n
    r(x(i))=1;
end
```