

Diplomarbeit

Welche Strategien nutzen Menschen zur Lösung des Traveling Salesperson Problems?

Erforschung von Pfadplanungsstrategien und Entwicklung neuer Algorithmen zur Vorhersage menschlichen Verhaltens bei der Lösung des TSPs

Nicole Nadine Ehbauer

Prüfer in der Biologie: Prof. Dr. H. A. Mallot

Prüfer in der Informatik: Prof. Dr. W. Rosenstiel

Betreuer: Jan M. Wiener

beendet am: 20. März 2006

Fakultät für Biologie

Zoologisches Institut
LS Kognitive Neurowissenschaft

Lehrstuhl Technische Informatik

Wilhelm-Schickard-Institut für Informatik
Eberhard Karls Universität Tübingen

Fakultät für Biologie
Zoologisches Institut
LS Kognitive Neurowissenschaft

Auf der Morgenstelle 28
D - 72076 Tübingen
Germany

Telefon: (+49/0) 7071 29 78830
(+49/0) 7071 29 78826 (Sekretariat)
Telefax: (+49/0) 7071 29 2891

E-Mail: hanspeter.mallot@uni-tuebingen.de
WWW: <http://www.uni-tuebingen.de/cog/>

Lehrstuhl für Technische Informatik
Eberhard-Karls-Universität Tübingen
Fakultät für Informatik

Sand 13/14
D-72076 Tübingen
Germany

Telefon: (+49/0) 7071 29 75482
(+49/0) 7071 29 78956 (Sekretariat)
Telefax: (+49/0) 7071 29 5062

E-Mail: rosenstiel@informatik.uni-tuebingen.de
WWW: <http://www-ti.informatik.uni-tuebingen.de>

Nicole Ehbauer
Quenstedtstraße 16/1
D-72076 Tübingen
Germany

Telefon: (+49/0) 7071 610421
E-Mail: Nicole.Ehbauer@gmx.de

ERKLÄRUNG

Hiermit versichere ich eidesstattlich, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Hilfsmittel verwendet habe.

Tübingen, den 20. März 2006

Nicole Ehbauer

Abstract

Abhängig von der Anzahl der Ziele und dem Layout der Umgebung kann die Planung eines Pfades ein sehr komplexes und rechenintensives Problem darstellen. Dies lässt sich eindrucksvoll an dem sogenannten Traveling Salesperson Problem (TSP) darstellen. Beim TSP besteht die Aufgabe darin, von einem gegebenen Ort ausgehend den kürzesten Pfad zu finden, welcher eine bestimmte Anzahl von Zielorten miteinander verbindet und zum Ausgangsort zurückkehrt. Möchte man 5 Ziele besuchen, so gibt es 24 verschiedene Anordnungen der zu besuchenden Ziele, wenn ein asymmetrisches TSP zu Grunde liegt. Ein asymmetrisches TSP besitzt den gleichen Anfangs- und Endpunkt. Somit bieten sich $(n - 1)!$ Möglichkeiten. Bei 8 Zielen sind es 5040, und bei 9 Zielen sind es schon 40320 Möglichkeiten.

Das TSP ist eine durchaus alltägliche Aufgabe. Zum Beispiel wird dieses Problem auf einer Einkaufsstrecke gelöst, auf der in verschiedenen Läden Einkäufe getätigt werden sollen. Offensichtlich werden dabei nicht alle möglichen alternativen Pfade berechnet und ein Vergleich der Längen verwendet um die kürzeste Variante zu finden, sondern es werden Strategien und Heuristiken angewendet, die es ermöglichen das Problem schnell und ohne hohen kognitiven Aufwand zu lösen. Die Strategien und Heuristiken, die Menschen bei solchen Planungsaufgaben anwenden, sind weitgehend unbekannt und Gegenstand dieser Arbeit.

Dependent on the amount of targets and the layout of the environment, planning a route can display a really complex and computational-intensely problem. That can be demonstrated by the so-called Traveling Salesperson Problem (TSP). In a TSP, a salesperson has a set of cities, which he wants to visit and by that detecting the shortest path over all cities and returning to the starting location. By visiting 5 cities there are 24 different sequences in which the cities are ordered, if it is an asymmetrical TSP. That means the starting and ending point is the same and so there are $(n - 1)!$ possibilities. By visiting 8 cities there are 5040 options and for 9 cities there are nearly 40320 different options.

The TSP is absolutely a everyday lesson which human, for example, solve on a shopping tour by visiting different stores doing some shopping. Obviously, human do not calculate all possible alternative routes on such a shopping tour and do not comparing their length to decide for the shortest alternative. In fact, they use strategies and heuristics which enable to solve the planning problem fast and without high cognitive costs. Strategies and heuristics which are used by human to solve such a planning problem are largely unknown and an item of this work.

GLOSSAR	
ABBILDUNGSVERZEICHNIS	
ALGORITHMENVERZEICHNIS	
TABELLENVERZEICHNIS	

Inhaltsverzeichnis

1	EINLEITUNG	21
2	THEORETISCHE GRUNDLAGEN AUS DER INFORMATIK	23
2.1	DAS TRAVELING SALESPERSON PROBLEM	23
2.1.1	KLASSISCHE DEFINITION	24
2.1.2	ALGORITHMISCHE KOMPLEXITÄT	25
2.1.3	DIE FAMILIE DER TSP PROBLEME	27
3	THEORETISCHE GRUNDLAGEN AUS DER BIOLOGIE	28
3.1	ARBEITSGEDÄCHTNIS (WORKING MEMORY)	28
3.1.1	DIE ZENTRALE EXECUTIVE (THE CENTRAL EXECUTIVE)	31
3.1.2	DIE PHONOLOGISCHE SCHLEIFE (THE PHONOLOGICAL LOOP)	32
3.1.3	DER VISUELL-RÄUMLICHER NOTIZBLOCK (THE VISUOSPATIAL SKETCHPAD)	33
3.1.4	DER EPISODISCHE PUFFER (THE EPISODIC BUFFER)	33
3.2	DAS LANGZEITGEDÄCHTNIS	35
3.3	DAS ORTSGEDÄCHTNIS	36
3.4	LERNEN	37
3.5	EINE STUDIE IM VORFELD	38
4	METHODEN UND MATERIALIEN	40
4.1	DER EXPERIMENTELLE AUFBAU	40
4.2	DIE NAVIGATIONSAUFGABE	41
4.2.1	PFADTYPEN	42
4.3	VERSUCHSPERSONEN	44
4.4	AUSWERTUNG	44
5	ERGEBNISSE EXPERIMENT 1	46
5.1	STARTZEIT	46
5.2	FEHLERQUOTE (<i>Overshoot</i>)	47
5.3	KORREKTE PFADE	49
5.4	ÜBERSCHREITUNGEN DER REGIONENGRENZEN	50

5.5	REGIONEN-BASIERTE STRATEGIE	51
5.6	BESTE UND SCHLECHTESTE PFADE	52
5.7	GEZEICHNETE PFADE ALLER VERSUCHSPERSONEN	52
5.8	KONTROLLFAKTOREN	53
5.9	AUSFALLPFADE	54
6	ERGEBNISSE EXPERIMENT 2	56
6.1	TRAININGSDATEN	56
6.2	AUSWERTUNG 1	57
6.2.1	STARTZEIT	57
6.2.2	FEHLERQUOTE (<i>Overshoot</i>)	58
6.2.3	KORREKTE PFADE	60
6.2.4	ÜBERSCHREITUNGEN DER REGIONENGRENZEN	61
6.2.5	REGIONEN-BASIERTE STRATEGIE	62
6.2.6	BESTE UND SCHLECHTESTE PFADE	63
6.2.7	GEZEICHNETE PFADE DER VERSUCHSPERSONEN	63
6.2.8	KONTROLLFAKTOREN	65
6.2.9	AUSFALLPFADE	65
6.3	AUSWERTUNG 2	65
6.3.1	STARTZEIT	65
6.3.2	FEHLERQUOTE (<i>Overshoot</i>)	66
6.3.3	KORREKTE PFADE	67
6.3.4	ÜBERSCHREITUNGEN DER REGIONENGRENZEN	68
6.3.5	REGIONEN-BASIERTE STRATEGIE	68
6.3.6	BESTE UND SCHLECHTESTE PFADE	69
6.3.7	KONTROLLFAKTOREN	69
6.3.8	AUSFALLPFADE	69
7	DISKUSSION	70
7.1	EXPERIMENT 1	70
7.2	EXPERIMENT 2	71
7.3	VERGLEICH DER BEIDEN EXPERIMENTE	73
8	VERGLEICH ZWISCHEN EXISTIERENDEN HEURISTIKEN UND MENSCHLICHER WEGEPLANUNG	74
8.1	NEAREST NEIGHBOUR ALGORITHMUS	74
8.2	NEAREST INSERTION ALGORITHMUS	76
8.3	RUBBER BAND ALGORITHMUS	76
8.4	METAHEURISTIKEN	78
8.4.1	SIMULATED ANNEALING	79
8.5	VERWENDETE STRATEGIEN DER VERSUCHSPERSONEN	83

9	ANSATZ EINES MODELLS FÜR DAS MENSCHLICHE VERHALTEN BEI DER LÖSUNG VON TSP-AUFGABEN	84
9.1	CLUSTER + NEAREST NEIGHBOUR ALGORITHMUS	84
9.1.1	IMPLEMENTIERUNG DES CLUSTER + NN ALGORITHMUS	85
9.2	NEAREST NEIGHBOUR ALGORITHMUS	87
9.2.1	IMPLEMENTIERUNG DES NEAREST NEIGHBOUR ALGORITHMUS	88
9.3	QUALITÄT DER ALGORITHMEN	89
9.3.1	CLUSTER + NN vs. NN	89
9.3.2	CLUSTER + NN vs. "MENSCH"	91
9.3.3	NN vs. "MENSCH"	92
10	ZUSAMMENFASSUNG UND AUSBLICK	94
11	LITERATUR	96
A	EXPERIMENT 1	100
A.1	ÜBERSCHREITUNGEN DER REGIONENGRENZEN	100
A.2	REGIONEN-BASIERTE STRATEGIE	101
B	EXPERIMENT 2	101
B.1	KORREKTE PFADE	101
B.2	ÜBERSCHREITUNGEN DER REGIONENGRENZEN	102
B.3	REGIONEN-BASIERTE STRATEGIE	104
C	FOTOS	105

Glossar

ACO - *Ant Colony Optimisation*; Optimierungsalgorithmus basierend auf Ameisen Kolonien

ACT-Theorie - *Adaptive Control of Thought*; Kognitive Architektur; Simulation von Denk- und Problemlöseprozessen

ANOVA - *Analysis of Variance*; beschreibt die Vergleichsmöglichkeiten verschiedener Stichproben

BTSP - *Bottleneck Traveling Salesperson Problem*

CE - *central executive*; zentrale Exekutive; Teil des Arbeitsgedächtnisses

EA - *evolutionary algorithms*; Evolutionäre Algorithmen

LS - *local search* ; Lokale Suche

LTM - *long term memory*; das Langzeitgedächtnis

MTSP - *Multi Traveling Salesperson Problem*

NI - *Nearest Insertion Algorithmus*

nichtregionale Pfadgruppe - Pfade, bei denen die Anwendung einer regionen-basierten Strategie nicht zu einem optimalen Pfad führt

NN - *Nearest Neighbour Algorithmus*

NP - *nondeterministic polynomial*; nichtdeterministisch polynomiell

P - *polynomial*; polynomiell

Pfadtyp 1 - siehe *regionale Pfadgruppe*

Pfadtyp 2 - siehe *nichtregionale Pfadgruppe*

regionale Pfadgruppe - Pfade bei denen die Anwendung einer regionen-basierten Strategie zu einem optimalen Pfad führen kann

SA - *Simulated Annealing Algorithmus*

SAT - *satisfiability*; Erfüllbarkeit der Aussagenlogik

STM - *short term memory*; das Kurzzeitgedächtnis

TS - *tabu search*; Tabu Suche

TSP - *Traveling Salesperson Problem*

WM - *working memory*; Working Memory Modell von Baddeley und Hitch; das Arbeitsgedächtnis

Abbildungsverzeichnis

- Abbildung 3* - Zusammenhänge zwischen Kurz- und Langzeitgedächtnis; Kapitel 3
- Abbildung 3.1a* - Working Memory Modell; Kapitel 3, Abschnitt 3.1
- Abbildung 3.1b* - visuelle Verarbeitung im Cortex; Kapitel 3, Abschnitt 3.1
- Abbildung 3.1.1* - System des Arbeitsgedächtnisses; Kapitel 3, Abschnitt 3.1.1
- Abbildung 3.2* - Einteilung des Langzeitgedächtnisses; Kapitel 3, Abschnitt 3.2
- Abbildung 3.1.4* - überarbeitetes Working Memory Modell; Kapitel 3, Abschnitt 3.1.4
- Abbildung 3.3* - Verbindungen im Cortex; Kapitel 3, Abschnitt 3.3
- Abbildung 3.4* - Lernen; Kapitel 3, Abschnitt 3.4
- Abbildung 4.1a* - Konfiguration des experimentellen Feldes; Kapitel 4, Abschnitt 4.1
- Abbildung 4.1b* - Die beiden Konfigurationen des experimentellen Feldes; Kapitel 4, Abschnitt 4.1
- Abbildung 4.2* - *shopping* Liste; Kapitel 4, Abschnitt 4.2
- Abbildung 4.4* - Beispiel für regionale und nichtregionale Pfade; Kapitel 4, Abschnitt 4.4
- Abbildung 5* - Beispiel für optimalen und suboptimalen Pfad; Kapitel 5
- Abbildung 5.1* - Startzeiten, Experiment 1; Kapitel 5, Abschnitt 5.1
- Abbildung 5.2a* - *Overshoot* der Pfadtypen, Zielpunkte und experimentellen Blöcke, Experiment 1; Kapitel 5, Abschnitt 5.2
- Abbildung 5.2b* - *Overshoot* der Pfadtypen, Experiment 1; Kapitel 5, Abschnitt 5.2
- Abbildung 5.3* - korrekt gefundene Pfade, Experiment 1; Kapitel 5, Abschnitt 5.3
- Abbildung 5.4* - Überschreitungen der Regionengrenzen, Experiment 1; Kapitel 5, Abschnitt 5.4
- Abbildung 5.5* - regionen-basierte Strategie, Experiment 1; Kapitel 5, Abschnitt 5.5
- Abbildung 5.7* - Pfade der Versuchspersonen, Experiment 1; Kapitel 5, Abschnitt 5.7
- Abbildung 6.1* - Anzahl der Fehler in den Trainingsläufen, Experiment 2; Kapitel 6, Abschnitt 6.1
- Abbildung 6.2.1* - Startzeiten, Experiment 2, Auswertung 1; Kapitel 6, Abschnitt 6.2.1
- Abbildung 6.2.2a* - *Overshoot* der Pfadtypen und Zielpunkte, Experiment 2, Auswertung 1; Kapitel 6, Abschnitt 6.2.2
- Abbildung 6.2.2b* - *Overshoot* der experimentellen Blöcke, Experiment 2, Auswertung 1; Kapitel 6, Abschnitt 6.2.2
- Abbildung 6.2.2c* - *Overshoot* der Pfadtypen, Experiment 2, Auswertung 1; Kapitel 6, Abschnitt 6.2.2
- Abbildung 6.2.3* - korrekt gefundene Pfade, Experiment 2, Auswertung 1; Kapitel 6, Abschnitt 6.2.3
- Abbildung 6.2.4* - Überschreitungen der Regionengrenzen, Experiment 2, Auswertung 1; Kapitel 6, Abschnitt 6.2.4
- Abbildung 6.2.5* - regionen-basierte Strategie, Experiment 2, Auswertung 1; Kapitel 6, Abschnitt 6.2.5
- Abbildung 6.2.7* - Pfade der Versuchspersonen, Experiment 2, Auswertung 1; Kapitel 6, Abschnitt 6.2.7
- Abbildung 6.3.2a* - *Overshoot* der Pfadtypen und Zielpunkte, Experiment 2, Auswertung 2; Kapitel 6, Abschnitt 6.3.2

Abbildung 6.3.2b - *Overshoot* der experimentellen Blöcke und der Pfadtypen, Experiment 2, Auswertung 2; Kapitel 6, Abschnitt 6.3.2

Abbildung 6.3.5 - regionen-basierte Strategie Experiment 2, Auswertung 2; Kapitel 6, Abschnitt 6.3.5

Abbildung 7 - Überblick über die Beziehungen im Arbeitsgedächtnis; Kapitel 7

Abbildung 8.1 - Beispiel für einen Pfad berechnet über den Nearest Neighbour Algorithmus; Kapitel 8, Abschnitt 8.1

Abbildung 8.2 - Beispiel für einen Pfad berechnet über den Nearest Insertion Algorithmus; Kapitel 8, Abschnitt 8.2

Abbildung 8.3 - Beispiel für den Graham Scan Algorithmus; Kapitel 8, Abschnitt 8.3

Abbildung 8.4 - Überblick über Metaheuristiken; Kapitel 8, Abschnitt 8.4

Abbildung 9 - Pfadbeispiele für Cluster + NN vs. NN; Kapitel 9

Abbildung 9.3.1a - *Overshoot*, Cluster + NN vs. NN; Kapitel 9, Abschnitt 9.3.1

Abbildung 9.3.1b - korrekt gefundene Pfade der Pfadtypen, Cluster + NN vs. NN; Kapitel 9, Abschnitt 9.3.1

Abbildung 9.3.1c - korrekt gefundene Pfade der Zielpunkte, Cluster + NN vs. NN; Kapitel 9, Abschnitt 9.3.1

Abbildung 9.3.1d - regionen-basierte Strategie, Cluster + NN vs. NN; Kapitel 9, Abschnitt 9.3.1

Abbildung A.1a - Überschreitungen der Regionengrenzen der Pfadtypen, Experiment 1; Anhang A, Abschnitt A.1

Abbildung A.1b - Überschreitungen der Regionengrenzen der Zielpunkte, Experiment 1; Anhang A, Abschnitt A.1

Abbildung B.1 - korrekt gefundene Pfade der Pfadtypen, Experiment 2, Auswertung 2; Anhang B, Abschnitt B.1

Abbildung B.2a - Regionenüberschreitungen zwischen den Pfadtypen, Experiment 2, Auswertung 1 und 2; Anhang B, Abschnitt B.2

Abbildung B.2b - Regionenüberschreitungen zwischen den Zielpunkten, Experiment 2, Auswertung 1 und 2; Anhang B, Abschnitt B.2

Foto C1 - experimentelle Umgebung; Anhang C

Foto C2 - Versuchsperson während der Navigation; Anhang C

Foto C3 - Versuchsperson während der Navigation; Anhang C

Algorithmenverzeichnis

Algorithmus 8.1 - Nearest Neighbour Algorithmus; Kapitel 8, Abschnitt 8.1

Algorithmus 8.3 - Graham Scan Algorithmus; Kapitel 8, Abschnitt 8.3

Algorithmus 8.4.1 - Simulated Annealing Algorithmus; Kapitel 8, Abschnitt 8.4.1

Algorithmus 9.1.1 - Implementierung des Cluster + NN Algorithmus; Kapitel 9, Abschnitt 9.1.1

Algorithmus 9.2.1 - Implementierung des NN Algorithmus; Kapitel 9, Abschnitt 9.2.1

Tabellenverzeichnis

Tabelle 4.2.1A - Zusammenfassung aller 36 Pfade

Tabelle 5.8A - Kontrollfaktor Feld; Experiment 1

Tabelle 5.8B - Kontrollfaktor Konfiguration; Experiment 1

Tabelle 6.2.2A - *Overshoot* zwischen männlichen und weiblichen Versuchspersonen; Experiment 2, Auswertung 1

Tabelle 6.2.8A - Kontrollfaktor Feld; Experiment 2, Auswertung 1

Tabelle 6.2.8B - Kontrollfaktor Konfiguration; Experiment 2, Auswertung 1

Tabelle 6.3.2A - durchschnittlicher *Overshoot* zwischen männlichen und weiblichen Versuchspersonen; Experiment 2, Auswertung 2

Tabelle 6.3.7A - Kontrollfaktor Feld; Experiment 2, Auswertung 2

Tabelle 6.3.7B - Kontrollfaktor Konfiguration; Experiment 2, Auswertung 2

Tabelle A.2A - durchschnittliche Häufigkeiten der Verwendung einer regionen-basierten Strategie zwischen weiblichen und männlichen Versuchspersonen; Experiment 1

Tabelle B.1A - durchschnittlicher Prozentsatz an korrekt gefundenen Pfaden zwischen männlichen und weiblichen Versuchspersonen; Experiment 2, Auswertung 2

Tabelle B.3A - durchschnittliche Häufigkeiten der Verwendung einer regionen-basierten Strategie zwischen weiblichen und männlichen Versuchspersonen, Experiment 2, Auswertungen 1 und 2

1 Einleitung

Das Traveling Salesperson Problem (TSP) wurde auf mathematischer Ebene von dem Irischen Mathematiker Sir William Rowan Hamilton (1805-1865) und seinem Britischen Kollegen Thomas Penyngton Kirkman schon im 19. Jahrhundert diskutiert. In dieser Zeit entwickelte W. Hamilton ein Spiel, das sogenannte "Hamilton's Icosian Game", und verkaufte die Rechte daran für 25 Pfund an einen Kaufmann aus London [TANLER, 2004]. Die heutige, als allgemeine Form bekannte, Definition des TSP wurde von dem Mathematiker Karl Menger in Wien und Harvard in den 1930ern studiert. Infolge dessen widmeten viele Wissenschaftler ihr Interesse diesem Problem, und so ist es noch heute.

Beim TSP besteht die Aufgabe darin, von einem gegebenen Startpunkt ausgehend den kürzesten Pfad zu finden, welcher eine bestimmte Anzahl von Zielpunkten miteinander verbindet und zum Ausgangspunkt zurückkehrt. Dabei steigt die Anzahl der möglichen Lösungen mit wachsender Anzahl an Zielpunkten.

Für Cristos Papadimitriou ist das Traveling Salesperson Problem weniger ein Problem, vielmehr eine Art Leidenschaft. Alle Wissenschaftler, die sich jemals mit diesem Problem beschäftigten, sind von ihm fasziniert. Erstens ist seine Formulierung einfach, anschaulich und selbsterklärend. Zweitens, das Problem bekämpft in Bezug auf seine NP-Vollständigkeit jeden Ansatz zur Entwicklung effizienter und optimaler Lösungsmethoden, sowie die Entwicklung einer konsistenten Approximationsmethode, welche annähernd gute Lösungen garantiert. Wissenschaftler, die mit der Entwicklung effizienter Heuristiken beschäftigt sind, sehen sich deshalb einer kaum zu bewältigenden Aufgabe gegenüber.

Auch wenn nur eine kleine Anzahl an Kauffleuten existiert, die an einer Lösung des Problems interessiert ist, gibt es doch eine große ökonomische Anzahl an ähnlichen Problemen, die auf das TSP zurückzuführen sind. So ist das Traveling Salesperson Problem auch als Einkaufs-, Boten oder Reiseproblem bekannt und ist eines der berühmtesten Probleme, sowie das am meist untersuchte kombinatorische Problem des 20. Jahrhunderts.

Die allgemeine Formulierung des TSPs ist verständlich für jeden Laien und in Kombination mit der Gegebenheit, dass dieses Problem nur schwer mit akzeptablen Kosten zu lösen ist, erlangte das TSP seine heutige Berühmtheit und dies nicht nur unter Mathematikern. Eine weitere Eigenschaft liegt darin, dass das TSP repräsentativ für eine ganze Reihe anderer ähnlicher kombinatorischer Probleme steht, welche genauso schwer zu lösen sind.

Das Traveling Salesperson Problem ist in keinster Weise nur ein interessantes Objekt für Mathematiker und Informatiker. In der Literatur finden sich Publikationen über dieses Thema von Physikern, Biologen und Wissenschaftlern aus vielen anderen Bereichen. Simulated Annealing (SA), zum Beispiel, ist eine Kombination von physikalischen Betrachtungen auf kombinatorische Probleme. Viele dieser Zusammenarbeiten führten zu der Entwicklung einiger bekannter Algorithmen, wie zum Beispiel die Evolutionären Algorithmen (EA), das vorher genannte Simulated Annealing, Ant Colony Optimisation (ACO) usw. Das TSP wird somit zur Entwicklung neuer Methoden in vielen verschiedenen Bereichen verwendet.

Der größte Nachteil des TSPs liegt jedoch in dem Problem, dass TSPs mit einer sehr hohen Anzahl an zu besuchenden Punkten, das heißt mehr als 20 zu besuchende Ziele, schwer zu lösen sind. Es werden komplexe Algorithmen benötigt, deren Implementierung ein großes theoretisches Wissen voraussetzt. Um dies zu gewährleisten sind die schnellsten Computer notwendig. Des Weiteren ist die benötigte Zeit zur Lösung des Problems sehr schwer vorherzusagen. Je größer das Problem, desto ineffizienter sind die Kosten.

Normalerweise lösen Menschen solche Planungsaufgaben schnell und effizient. Anstelle alle möglichen Pfade zu berechnen und miteinander zu vergleichen, vertrauen menschliche Navigatoren auf Strategien und Heuristiken, welche den kognitiven Aufwand minimieren und dennoch annähernd gute Lösungen ergeben.

Es gibt einige Publikationen an Studien, welche die Qualität von Lösungen des TSPs am Menschen erforschten. Zum Beispiel, haben J. N. MacGregor and T. Ormerod ein Experiment durchgeführt, in welchem sie 10-Punkt- und 20-Punkt-Probleme verwendeten. Jede Versuchsperson bekam eine

ausgewählte Anzahl an Problemen und Instruktionen, welche die Auswahl des Startpunktes und das Zeichnen des von ihnen gewählten kürzesten Pfades beinhaltete. Dieser Pfad sollte jeden Punkt genau ein einziges Mal passieren und zum Startpunkt zurückkehren. Nach Durchführung des Experiments zeigten die Ergebnisse eine überraschend hohe Qualität der menschlichen Lösungen, im Vergleich mit den Lösungen bekannter Algorithmen [MACGREGOR, 1996].

Pfadplanung und Pfadwahl wurde in Tieren und Menschen in nur wenigen Studien erforscht und das Wissen über die zugrunde liegenden Mechanismen, Strategien und Heuristiken, sowie der beteiligten kognitiven Komponenten und Prozesse, ist lückenhaft.

Um die Mechanismen und Strategien, welche an der Pfadplanung beteiligt sind, zu erforschen, ist es wichtig die beteiligten Prozesse und Komponenten zu betrachten. Das in dieser Arbeit angenommene Schema der Pfadplanung erfolgt in 4 Schritten. Im *ersten Schritt* werden die Positionen der Zielpunkte in der Umgebung lokalisiert. Im *zweiten Schritt* wird darauf eine Darstellung im Arbeitsgedächtnis aufgebaut, die die aktuelle Position, die Positionen der Zielpunkte und den dazugehörigen Teilbereich der Umgebung enthält. Innerhalb dieser Darstellung wird der aktuelle Pfadplan generiert und im *dritten Schritt* im Gedächtnis behalten. Im *vierten Schritt* wird der Pfadplan umgesetzt. Dieses Schema kann in seiner Grundform auf einfache Planungsaufgaben übertragen werden. Einfache Planungsaufgaben entstehen beispielsweise anhand von kleinen Umgebungen, in denen nur wenige Pfadvarianten vorkommen können, sowie bei entstehenden Pfadplänen, die einfach und leicht im Gedächtnis zu behalten sind. Bei steigender Schwierigkeit der Planungsaufgaben muss eine Modifizierung des Schemas erfolgen.

Welche Strategien verwenden Menschen um das Traveling Salesperson Problem zu lösen? Dafür wurden im empirischen Teil dieser Arbeit Versuchspersonen gebeten in einem großen Raum, welcher verschiedene Zielpunkte enthielt, den kürzesten Pfad zu finden. Die Anordnung der 25 verschiedenen Zielpunkte im Feld ist konstant und das Feld wurde dabei zur Vereinfachung der Aufgaben in fünf Regionen eingeteilt. Die Komplexität der Aufgaben wird systematisch variiert, indem die Anzahl der Zielpunkte, welche zu besuchen sind, verändert wird. Es gibt Aufgaben in welcher eine regionen-basierte Strategie, das heißt eine Strategie, welche Region für Region besucht, zu einer optimalen Lösung führen kann. Es gibt jedoch auch Aufgaben, bei denen diese Strategie nicht funktioniert und der gefundene Pfad suboptimal verläuft. Nun soll erforscht werden, in wie weit eine regionen-basierte Strategie Verwendung findet und welche Qualität die gefundenen Lösungen aufweisen.

Im zweiten Teil dieser Arbeit soll untersucht werden, in welcher Weise sich die Strategien der Versuchspersonen ändern, indem nicht nur das Arbeitsgedächtnis, sondern auch das Ortsgedächtnis eine Rolle bei der Lösung der Aufgaben spielt. Die 25 Symbole auf dem Feld werden dabei abgedeckt und sind für die Versuchspersonen während der Navigation nur sichtbar, wenn diese von den Versuchspersonen aufgedeckt werden. Bei den zu lösenden Aufgaben soll hier nicht nur der kürzeste Weg über die zu besuchenden Zielpunkte gefunden werden, sondern es soll auch die Anzahl an falsch aufgedeckten Symbolen so niedrig wie möglich gehalten werden. Um dies zu gewährleisten, durchliefen die Versuchspersonen vor der Lösung der TSP-Aufgaben eine experimentelle Trainingsphase um sich die richtige Orientierung der Zielpunkte so gut wie möglich einzuprägen. Nach der Trainingsphase, sollte es den Versuchspersonen möglich sein den kürzesten Pfad, bei verdecktem Feld mit so wenig Fehlern wie möglich, zu finden.

Nach den beiden oben beschriebenen Experimenten, wurden die Ergebnisse der Lösungen von Menschen mit dem zuvor über die Euklidische Distanz berechneten kürzesten Pfad verglichen und es wurde versucht einige Vorhersagen über Strategien zu machen, welche Menschen verwenden um das Wegeproblem zu lösen. Auch wurde versucht einen Algorithmus zu implementieren, welcher menschliches Verhalten in Bezug auf das TSP vorhersagen kann.

2 Theoretische Grundlagen aus der Informatik

In diesem Kapitel sollen die theoretischen Grundlagen der Informatik behandelt werden. Zum einen ist es wichtig einen Blick auf die theoretischen Aspekte des Traveling Salesperson Problems zu werfen, da sie einen Einblick in die grundlegenden Eigenschaften des TSPs geben. Dabei wird die klassische Definition und einige Aspekte der Komplexität, sowie die NP-Vollständigkeit vorgestellt.

2.1 Das Traveling Salesperson Problem

Das Traveling Salesperson Problem, einstmals bekannt als das Taveling Salesman Problem, ist ein weit verbreitetes Problem in der Theoretischen Informatik. In seiner euklidischen Form soll der kürzeste Pfad um eine bestimmte Menge an Zielpunkten gefunden werden, wobei jeder Punkt nur einmal besucht werden soll und der Pfad zum Startpunkt zurückführt. Dabei sollen gleichzeitig die Kosten so niedrig wie möglich gehalten werden. Die Lösung stellt somit einen kompakten Pfad dar.

Der Mathematiker K. Menger hat sich mit seiner Arbeit dem Traveling Salesperson Problem gewidmet. Der Anfang seines Interesses kommt in seiner Veröffentlichung "*Ein Theorem über die Botenlänge*" im Jahr 1928, sowie in der im Jahre 1929 folgenden Veröffentlichung "*Eine weitere Verallgemeinerung des Längenbegriffs*", zum Ausdruck. Darin studierte er die Länge $l(C)$ einer einfachen Kurve C in einem metrischen Raum S .

Allgemein beschreibt Menger das TSP wie folgt:

"Wir bezeichnen als Botenproblem (weil diese Frage in der Praxis von jedem Postboten, übrigens auch von vielen Reisenden zu lösen ist) die Aufgabe, für endlichviele Punkte, deren paarweise Abstände bekannt sind, den kürzesten die Punkte verbindenden Weg zu finden. Dieses Problem ist natürlich stets durch endlich viele Versuche lösbar. Regeln, welche die Anzahl der Versuche unter die Anzahl der Permutationen der gegebenen Punkte herunterdrücken würden, sind nicht bekannt. Die Regel, man solle vom Ausgangspunkt erst zum nächstgelegenen Punkt, dann zu dem diesem nächstgelegenen Punkt gehen usw., liefert im allgemeinen nicht den kürzesten Weg."¹

Menger war sich der Komplexität des TSPs bewusst, auch wusste er zu dieser Zeit bereits, dass die bekannte Nearest Neighbour Heuristik nicht immer eine optimale Lösung finden kann [SCHRIJVER].

Das TSP ist ein klassisches graphentheoretisches Problem. Es umfasst eine reisende Handelsperson, welche jede Stadt in einem gegebenen Netzwerk einmal besuchen soll, bevor sie zu ihrem Ausgangspunkt zurückkehrt und damit die Rundreise vollständig abschließt. Das Ziel ist die Rundreise zu finden, welche die wenigsten Kosten verursacht. Das heißt, den kürzesten Weg zu finden, welcher jede Stadt genau einmal besucht und dann zum Ausgangspunkt zurückkehrt und somit die zurückgelegte Distanz minimiert [HAUSKRECHT, 2005].

Liegt eine graphentheoretische Terminologie zu Grunde, so werden die Städte als Knoten und die Wege zwischen den Städten als Kanten dargestellt. Eine mögliche Rundreise, welche nicht notwendigerweise optimal sein muss, ist dann eine zyklische zusammenhängende Reihe von Kanten, welche jeden Knoten genau einmal berührt. Jeder Knoten ist vom Rang 2, das heißt, dass er genau eine eingehende und eine ausgehende Kante besitzt. Solch eine Reihe von Kanten wird Hamiltonkreis genannt. Die optimale Lösung ist somit der Hamiltonkreis mit der minimalsten

¹ "We denote by messenger problem (since in practice this question should be solved by each postman, anyway also by many travelers) the task to find, for finitely many points whose pairwise distances are known, the shortest route connecting the points. Of course, this problem is solvable by finitely many trials. Rules which would push the number of trials below the number of permutations of the given points, are not known. The rule that one first should go from the starting point to the closest point, then to the point closest to this, etc., in general does not yield the shortest route."

Länge aller möglichen Lösungen.

Liegt ein kompletter und ungerichteter Graph vor, in welchem die Verbindung zwischen jeder Stadt unabhängig von ihrer Richtung ist, so gibt es kein Problem bei der Ermittlung eines Hamiltonkreises. Es ist hierbei in jedem Fall gewährleistet, dass am Ende ein Hamiltonkreis existiert und somit wird der Graph als hamiltonisch bezeichnet.

Nach dieser Charakterisierung sollen nun spezifische Probleme bei der Lösung des TSP behandelt werden. Es werden zumeist verschiedene Typen des klassischen TSPs unterschieden, die in Abhängigkeit der Eigenschaften ihrer Distanzmatrix $d_{i,j}$ in verschiedene Bereiche gegliedert werden.

Definition 1: Ein TSP ist *symmetrisch*, wenn für alle Distanzen $d_{i,j} = d_{j,i}$ ($i, j \in \{1, 2, \dots, n\}$, $i \neq j$) gilt.

Aus dieser Definition folgt, dass die Distanz zwischen zwei Städten unabhängig von der Richtung ihrer Kante ist. Mittels der Symmetrie der Distanzmatrix ist es nur nötig die obere oder untere Dreiecksmatrix zu erstellen um alle Distanzen zu erhalten.

Definition 2: Gilt die obere Definition nicht für alle Distanzen in einem TSP, so ist das TSP *asymmetrisch*.

In Bezug auf die oberen beiden Definitionen ist die Anzahl der möglichen Rundreisen in einem *asymmetrischen* TSP $(n - 1)!$, und für ein *symmetrisches* gilt $\frac{(n-1)!}{2}$. Somit steigt die Anzahl möglicher Lösungen im Fall eines wachsenden n rapide an. Also ist dieses Problem nur effizient lösbar, wenn ein deterministischer Algorithmus existiert, welcher das Problem in polynomialer Zeit lösen kann [SCHMITTING, 1999].

Aus den vielen verschiedenen existierenden Definitionen des TSPs ist die klassische Definition die wohl am Bekannteste. Diese soll im folgenden Abschnitt beschrieben werden.

2.1.1 Klassische Definition

Definition 3: In der klassischen Definition des TSPs sind n Städte gegeben. Diese sind über nicht-negativ gewichtete Kanten verbunden und vollständig vernetzt. Nun soll der kostengünstigste Pfad für eine Rundreise, bei der alle Knoten genau einmal berührt werden und die zurück zum Anfangspunkt führt, gefunden werden.

Für eine algorithmische Lösung ist es vorteilhaft das Problem mit Hilfe eines Graphen zu lösen um damit einen Hamiltonkreis zu finden. Deshalb wird eine Menge $\{c_1, c_2, \dots, c_N\}$ von Zielorten vorgegeben und für jedes Paar $\{c_i, c_j\}$ an eindeutigen Zielorten eine Distanz $d(c_i, c_j)$ berechnet. Ziel hierbei ist eine Folge π der Zielorte zu finden, welche die Formel

$$\left(\sum_{i=1}^{N-1} d(c_{\pi(i)}, c_{\pi(i+1)}) \right) + d(c_{\pi(N)}, c_{\pi(1)}) \quad (2.1.1)$$

minimiert.

Die Formel 2.1.1 ist als die Weglänge charakterisiert, da sie die Länge des Pfades darstellt den ein Handelsmann gehen würde, wenn er die Zielpunkte in der Reihenfolge abläuft, welche mittels einer Permutation bestimmt wurde und am Ende wieder zum Ausgangspunkt zurückkehrt [JOHNSON, 1995]. Die Distanzmatrix $d(i, j)$ enthält hierbei die Distanzen zwischen den Paaren von Städten. Betrachtet man einen gewichteten Graphen, so ist die Bewertung jeder Kante eine positive reelle Zahl. Die Kanten des Graphen repräsentieren die Städte und die Kanten entsprechen reell gewichteten Pfadlängen zwischen den Städten. Die Länge der Kanten muss hierbei nicht der euklidischen oder der allgemeinen metrischen Distanz zwischen zwei verbundenen Knoten entsprechen. Das bedeutet, die Dreiecksungleichung muss hierfür nicht gelten.

2.1.2 Algorithmische Komplexität

Worin liegt das Problem, dass es so schwer macht das Traveling Salesperson Problem exakt zu lösen? Das TSP ist Teil einer Reihe von Problemen, welche NP-vollständig sind. NP bedeutet *nichtdeterministisch polynomial*. Einfacher gesagt, wenn wir eine Straße entlang gehen und es ein Orakel gäbe, das uns zu jedem beliebigen Zeitpunkt sagen könnte, welche Weg wir einschlagen müssen um am Ende den kürzest möglichen Weg zu erhalten, hätten wir die Möglichkeit jedes Problem exakt zu lösen. So ein Orakel steht bis heute noch nicht zur Verfügung und somit sind solche exakten Methoden in der Praxis nicht von Bedeutung.

Das TSP kann in endlicher Zeit gelöst werden, denn es ist ein endliches Problem. Bei einer großen Anzahl an Zielpunkten kann eine lange Zeit vergehen bis die exakte Methode auch zu einer exakten Lösung führt.

Bei einem geschlossenen Graphen mit *asymmetrischem* TSP und n Knoten existieren $(n - 1)!$ mögliche Lösungen. Dies liegt an der variablen Reihenfolge der Zielpunkte innerhalb des konstant bleibenden Anfangs- und Endpunktes (in diesem Fall ist der Anfangs- gleich dem Endpunkt) [TANLER, 2004].

Die Idee der NP-Vollständigkeit stammt aus der Komplexitätstheorie, einem Teil der Theoretischen Informatik. Sie wurde 1971 von Stephen Cook eingeführt im Zusammenhang mit dem von ihm vorgeschlagenen "*Cook Theorem*". In diesem Theorem wird eine spezielle formale Sprache der Komplexitätsklasse NP beschrieben. In dieser Klasse von Problemen sind korrekte Lösungen schwer zu finden, jedoch einfach zu prüfen. Cook entwickelte unter Verwendung des logistischen Modells einer nichtdeterministischen Turing-Maschine einen komplexen Beweis, dass das Erfüllbarkeitsproblem (satisfiability) NP-vollständig ist.

Definition 4: "*Cook Theorem*": SAT, die Erfüllbarkeit der Aussagenlogik, ist NP-vollständig.

Das heißt, dass eine effiziente Lösung des Erfüllbarkeitstheorems zu einer Lösung aller Optimierungs- und Entscheidungsprobleme in NP führt in Abhängigkeit der Größe der polynomiellen Anzahl an grundlegenden arithmetischen Operationen.

Definition 5: NP-vollständige Sprachen haben zwei allgemeine Eigenschaften.

1. Bis heute existiert für keine Entscheidungsmethode der Beweis, dass sie die korrekte Lösung in polynomieller Zeit liefern kann.
2. Wenn ein Algorithmus für eine NP-vollständige Sprache gefunden wird, und dieser in polynomieller Zeit eine korrekte Lösung liefert, so können alle Probleme der Klasse NP in polynomieller Zeit gelöst werden.

Das TSP ist NP-hart und so hat jeder Algorithmus für das Finden optimaler Pfade eine worst-case Laufzeit, welche in größerem Maße als jede polynomielle Laufzeit ansteigt (vorausgesetzt es gilt die allseitig bekannte Annahme $P \neq NP$). Dies führt zu zwei Möglichkeiten. Entweder zu der Entwicklung von Heuristiken, die schnell, jedoch nur annähernd optimal sind. Oder aber, man versucht Optimierungsalgorithmen zu entwickeln, welche gut in "Echtzeit" arbeiten [JOHNSON, 1995].

Jede feste Instanz des TSPs hat eine begrenzte Menge an Lösungen. Deshalb ist eine Methode für die Problemlösung eine komplette Suche im Lösungsraum. Eine vollständige Suche jedoch wird zunehmend unpraktischer, je höher die Anzahl an Zielpunkten n ansteigt. Dies folgt aus der Anzahl an möglichen Lösungen, die mit $\frac{(n-1)!}{2}$ (unter Vernachlässigung der Richtung des Pfades) berechnet wird. Die Anzahl der Lösungen steigt hierbei mit wachsendem n .

Definition 6: Ein Problem L ist NP-vollständig genau dann, wenn

1. L Teil der Klasse NP ist, das heißt, $L \subset NP$, und
2. L NP-hart ist, das heißt, jedes Problem in NP kann auf L reduziert werden.

Das Konzept der NP-Vollständigkeit ist nur für Entscheidungsprobleme definiert. Bei solchen Problemen, die auf dem Weltproblem der formalen Sprachen basieren, werden Fragen nur mit *JA* oder *NEIN* beantwortet [SCHMITTING, 1999].

Die Theorie der NP-Vollständigkeit ist zum einen eine Klassifizierung von Problemen infolge der Komplexität algorithmischer Lösungen, zum anderen eine Klassifikation aufgrund der bestbekanntesten Algorithmen für ein Problem, sowie eine Klassifikation des asymptotischen Verhaltens eines Algorithmus. Die Einschränkung der NP-Vollständigkeit beruht auf der Zeitkomplexität, der worst-case Analyse des Algorithmus und einzig auf Entscheidungsproblemen (Probleme, welche "ja" oder "nein" liefern). Die Zeitkomplexität wird gemessen an der Anzahl von Grundbefehlen und die Formalisierung richtet sich nach der $\mathcal{O}(\circ)$ -Notation.

Definition 7: Sei $f, g : \mathbb{N} \rightarrow \mathbb{N}$, so ist $f(n) \in \mathcal{O}(g(n))$, wenn eine positive natürliche Zahl c und c_0 existiert, dass für alle $n \geq n_0$ die Formel $f(n) \leq c \cdot g(n)$ gilt.

Ein Algorithmus ist polynomisch, wenn eine Zeitkomplexität $\mathcal{O}(p(n))$ existiert, wobei $p(n)$ ein Polynom ist. Ein Algorithmus ist exponentiell, wenn die Zeitkomplexität nicht polynomiell begrenzt ist.

Grundklassen sind die Klasse P, in welcher Entscheidungsprobleme mit polynomiellen Algorithmen gelöst werden können, und die Klasse NP, in welcher Entscheidungsprobleme mit nichtdeterministisch polynomiellen Algorithmen gelöst werden können. Ein nichtdeterministischer Algorithmus ist eine nichtdeterministische Turing-Maschine. Eine Turing-Maschine schätzt in der ersten Phase die Lösung ab um in der zweiten Phase die zuvor geschätzte Lösung in polynomieller Zeit deterministisch nachzuprüfen. Es ist bewiesen, dass $P \subseteq NP$, aber nicht, dass $P = NP$ ist. Somit ist $P \neq NP$ eine heute allgemein akzeptierte Annahme.

Die polynomische Redundanz wird in der nachfolgenden Definition 8 erklärt.

Definition 8: Ein Problem Π ist polynomiell reduzierbar auf ein Problem Π' , wenn ein polynomieller Algorithmus existiert, so dass jede Instanz von Π auf eine Instanz von Π' reduziert werden kann und für jede Instanz von Π ein "ja" ausgegeben wird, genau dann, wenn "ja" für die entsprechende Instanz von Π' ausgegeben wird.

Mit Bezug auf die oben erwähnte Definition 8 wird die NP-Vollständigkeit im Folgenden beschrieben.

Definition 9: Ein Problem Π ist NP-vollständig, genau dann, wenn zum einen Π in NP liegt und zum anderen für alle Π' in NP gilt, dass Π' polynomiell reduzierbar auf Π ist.

Beachtet werden muss, dass die Klasse der NP-vollständigen Probleme die schwierigsten Probleme in NP umfassen. Sollte für eines dieser Problem ein polynomieller Algorithmus gefunden werden, führt dies dazu, dass $P = NP$ ist. Bis heute ist es jedoch sehr unwahrscheinlich einen polynomiellen Algorithmus für ein NP-vollständiges Problem zu finden.

Neben der NP-Vollständigkeit gibt es Probleme, die NP-hart sind. NP-harte Probleme sind wie folgt definiert.

Definition 10: Ein Problem ist NP-hart, genau dann, wenn für alle Π' in NP gilt, dass Π' polynomiell reduzierbar auf Π ist.

Somit sind viele kombinatorische Optimierungsprobleme NP-hart.

Liegt ein Problem in der Klasse NP (*nichtdeterministisch polynomiell*), so existiert ein nichtdeterministischer Algorithmus, welcher das Problem in polynomieller Zeit löst.

Jedes NP-vollständige Problem ist reduzierbar auf jedes andere NP-vollständige Problem. Wird ein Algorithmus gefunden, welcher jedes einzelne NP-vollständige Problem in polynomieller Zeit löst, so existiert ein polynomieller Algorithmus für jedes NP-vollständige Problem und diese Probleme sind nicht als NP-hart klassifiziert.

Nachdem alle wichtigen Begriffe zum theoretischen Hintergrund des TSPs geklärt wurden, lässt sich in Hinblick auf das TSP eine kurze Zusammenfassung geben.

Wie oben schon erwähnt, gehören alle Probleme für die ein effizienter Algorithmus existiert, zur

Klasse P. Das heißt, diese Probleme können in polynomieller Zeit gelöst werden. Alle anderen Probleme gehören zur Klasse NP, der Klasse der nichtdeterministisch polynomiell lösbaren Probleme.

Das TSP gehört zur Klasse NP, aber nicht zur Klasse P. Somit ist es mit Hilfe eines nichtdeterministischen Algorithmus und einer polynomiellen Anzahl an Schritten optimal lösbar. Dies wird lediglich durch die Größe des Problems begrenzt. Ein deterministischer Algorithmus entspricht der klassischen Definition eines Algorithmus.

Definition 11: Die klassische Definition eines Algorithmus besagt, dass in jedem Schritt der nachfolgende Schritt über eine eindeutige Regel definiert ist.

Ein nichtdeterministischer Algorithmus hat somit die Möglichkeit die richtige Alternative aus mehreren Möglichkeiten zu wählen, um die optimale Lösung zu erhalten. In der Realität ist dieser Algorithmus nicht möglich, denn er würde die Fähigkeit besitzen jedes Problem effizient zu lösen. Dies wiederum würde dazu führen, dass jedes Problem der Klasse P auch in NP liegt, da die Probleme optimal mit Hilfe eines nichtdeterministischen Algorithmus gelöst werden können. Diese Frage ist heute noch unbeantwortet, da bisher noch niemand weiß, ob $P = NP$ ist. Daher gilt die allgemeine Annahme, dass $P \neq NP$ ist.

Es existiert eine Reihe von Optimierungsproblemen, welche nach bisherigem Stand in NP liegen, jedoch nicht in P. Diese Optimierungsprobleme sind schwierig zu lösen und genau dies gilt auch im Falle des TSP.

Des Weiteren kann der Menge an Problemen in NP eine Untermenge zugeordnet werden, in der die NP-vollständigen Probleme liegen. Für diese Probleme ist unbekannt, ob sie zur Klasse P gehören. Bis heute gibt es keinen effizienten Algorithmus zu deren Lösung. Es gilt jedoch, dass jedes dieser Probleme in die anderen Probleme überführt werden kann. Dies festigt wiederum den Beweis, dass sobald eines dieser Probleme in P liegt und ein effizienter Algorithmus zur Lösung existiert, jedes andere NP-vollständige Problem in P liegt und damit effizient lösbar ist [SCHMITTING, 1999].

2.1.3 Die Familie der TSP Probleme

Die im vorherigen erwähnte Existenz mehrerer Definitionen für das TSP lässt eine Unterscheidung der TSP-Probleme in die klassische Form und in die Familie von TSP-Problemen zu.

Eine Modifizierung des TSPs führt beispielsweise zu dem Multi Traveling Salesperson Problem (MTSP). Beim MTSP sucht nicht nur "ein" Handlungsreisender den kürzesten Pfad. Hier versuchen m Handlungsreisende den optimalen Pfad zu finden. Diese Reisenden können alle zur Lösung des Problems herangezogen werden ("non-lazy salesperson"), oder die Reisenden werden nur für die Lösung von Teilstücken des Problems verwendet. Zum Schluss berechnet sich der komplette Pfad aus der minimierten Summe über alle Pfade jedes Reisenden. Das MTSP ist wiederum reduzierbar auf das klassische TSP.

Beim Bottleneck TSP (BTSP; bottleneck, zu deutsch: Flaschenhals) verändert sich die Zielfunktion. Der komplette Pfad wird nicht mehr minimiert, sondern die längste vom Pfad berührte Kante (die Distanz zwischen zwei Städten) soll minimal werden. Das BTSP ist unter speziellen Bedingungen in polynomieller Zeit lösbar.

Diese beiden Probleme sind nur zwei Beispiele aus der großen Vielfalt der TSP-Familie. Viele dieser Probleme sind auf das klassische TSP zurückzuführen, andere sind eigenständige Mitglieder der TSP-Familie. Einige dieser eigenständigen Mitglieder weisen Spezialfälle auf, welche in polynomieller Zeit lösbar sind, andere jedoch sind NP-vollständig [SCHMITTING, 1999].

3 Theoretische Grundlagen aus der Biologie

In diesem Kapitel sollen die theoretischen Grundlagen der Biologie behandelt werden. Dabei soll vor allem auf die Beschreibung des Arbeitsgedächtnisses mit seinen Komponenten und auf das Ortsgedächtnis eingegangen werden, da diese für das Verstehen der menschlichen Komponente dieser Arbeit benötigt wird.

Das Gedächtnis ist die Fähigkeit des Gehirns Informationen aufzunehmen, zu speichern, zu ordnen und wieder abzurufen. Je nach zeitlicher Dauer der Speicherung von Information wird zwischen dem Kurzzeitgedächtnis und dem Langzeitgedächtnis unterschieden. Das Modell des Kurzzeitgedächtnisses wurde in den letzten Jahren vom Modell des Arbeitsgedächtnisses (Working Memory) abgelöst.

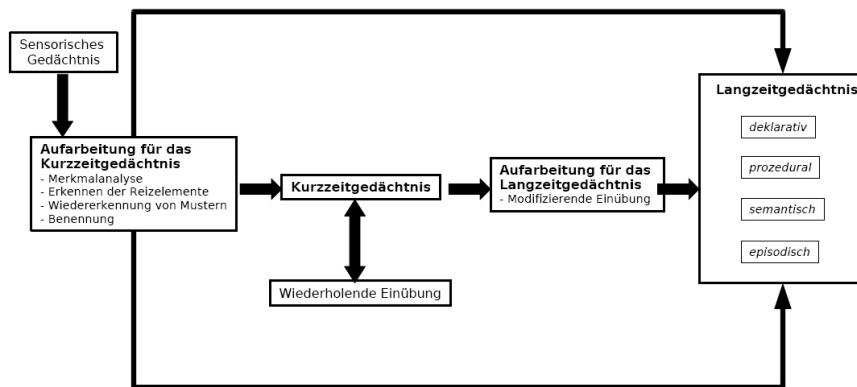


Abbildung 3: Zusammenhänge zwischen Kurz- und Langzeitgedächtnis [AMANN].

Das Gedächtnis ist nicht nur thematisch, sondern auch zeitlich organisiert. Eine neue Information wird zuerst im Arbeitsspeicher des Kurzzeitgedächtnisses, auf den wir schnellen Zugriff haben, abgelegt (Abbildung 3). Aus Gründen der Sparsamkeit muss dieser jedoch immer wieder von Ballast befreit werden. Dabei werden wichtige Informationen in den Mittel- oder Langzeitspeicher verschoben und Unwichtiges gelöscht. Je häufiger eine Information aufgerufen wird, desto genauer wird die Rückmeldung und somit auch die Beurteilung und Optimierung.

3.1 Arbeitsgedächtnis (Working Memory)

"The term working memory implies a system for the temporary holding and manipulation of information during the performance of a range of cognitive tasks such as comprehension, learning and reasoning." (Das Arbeitsgedächtnis nach Baddely)²

² "Der Begriff des Arbeitsgedächtnisses beinhaltet ein Verfahren für das zeitliche Festhalten und Beeinflussen von Information während der Ausführung einer Reihe von kognitiven Aufgaben, wie zum Beispiel verstehen, lernen und logisches Denken."

Das Gedächtnis besteht aus dem Ultra-Kurzzeitgedächtnis, den mittelfristigen Gedächtnisstrukturen und dem Langzeitgedächtnis. Zu den mittelfristigen Gedächtnisstrukturen gehören das Kurzzeitgedächtnis, beziehungsweise das Arbeitsgedächtnis (Working Memory).

Das Kurzzeitgedächtnis wurde 1960 im Modell von Atkinson und Shiffrin vorgestellt. Es ist ein Gedächtnissystem mit Registerstrukturen und bestimmten Eigenschaften. Zum einen hat es eine begrenzte Speicherdauer (ungefähr 15 - 30 Sekunden), sowie eine begrenzte Aufnahme an Speichermenge (ungefähr 7 ± 2 Informationseinheiten). Menschen unterscheiden sich in der Kapazität dieses Speichers (Miller's magische Zahl 7). Mit Aufmerksamkeit versehene Informationen, das heißt, Informationen über die man sich bewusst ist, werden in ein zwischen geschaltetes Kurzzeitgedächtnis überführt, und dort memoriert, bevor sie in das Langzeitgedächtnis gelangen. Das Langzeitgedächtnis besitzt eine relativ lange Speicherdauer, jedoch können neue Informationen die Alten verdrängen. Die Lebensdauer einer Information kann beispielsweise mittels Übung verlängert werden. Hierzu führten Shepard und Teghtsoonian eine Studie durch, in der den Versuchspersonen eine lange Sequenz von 200 dreistelligen Zahlen gezeigt wurde. Die Aufgabe bestand darin, zu erkennen, ob eine Zahl zum wiederholten Male vorkam. Es war von Interesse den Abstand zwischen der Zahlenwiedererkennung festzustellen, in Abhängigkeit, wie viele Zahlen zwischen der ersten und wiederholten Erkennung lagen. Shepard und Teghtsoonian nahmen an, dass das Erinnern zunehmend schlechter ausfallen müsste, sobald die Zahlen aus dem Kurzzeitgedächtnis verdrängt werden.

Infolge dieser und mehrerer anderer Studien zum Thema Kurzzeitgedächtnis kamen am bisherigen Modell etliche Zweifel auf. Die Kapazität schien weit geringer und auch die Gedächtnisspanne schien von anderen Faktoren abhängig zu sein.

Das Kurzzeitgedächtnis wurde deshalb 1974 vom Arbeitsgedächtnis abgelöst und von Baddeley und Hitch in ihrem Working Memory Modell beschrieben. Das Arbeitsgedächtnis entspricht in seinen Eigenschaften dem Kurzzeitgedächtnis.

Definition 12: Das Arbeitsgedächtnis ist ein System für die Verarbeitung und Speicherung von Informationen und der Koordination der dafür erforderlichen Ressourcen [KÖHLER/MEINEL].

Im Arbeitsgedächtnis werden sprachbasierte Informationen, die in einer bestimmten Zeitdauer memoriert werden können, gespeichert und manipuliert. Beim ausführlichen Abspeichern von Informationen arbeiten Kurzzeitgedächtnis und Arbeitsgedächtnis zusammen. Dabei ist das Kurzzeitgedächtnis ein Puffer für das Arbeitsgedächtnis und es findet zwischen dem Arbeitsgedächtnis und dem Langzeitgedächtnis ein reger Austausch an Informationen statt. Die Kapazität des Arbeitsgedächtnisses ist, wie das des Kurzzeitgedächtnisses, begrenzt. Dabei stellen sogenannte *Chunks* die Einheiten des Arbeitsgedächtnisses dar. Der Vorteil von *Chunks* liegt zum einen in der Reduktion der Anzahl der Informationen (*Items*) im Arbeitsgedächtnis, zum anderen in der Assoziation mit dem Langzeitgedächtnis. Bei reduzierter Anzahl an *Items* erfolgt ein höherer Transfer in das Langzeitgedächtnis [KÖHLER/MEINEL]. Wie die Kapazität, ist auch die zeitliche Dauer des Verweilens einer Information im Arbeitsgedächtnis begrenzt. Auf diese Dauer haben vor allem die Aufmerksamkeit für eine Information, sowie die Ähnlichkeit zu anderen Informationen einfluss.

Baddeley (1986) vertritt die Meinung, dass für den Umfang der Gedächtnisspanne die Geschwindigkeit mit der das Lernmaterial memoriert werden kann, entscheidend ist und nicht die Anzahl der Elemente. Dabei vergleicht er die Leistung, die man erbringen muss um Informationen im Kurzzeitgedächtnis zu halten, mit der Leistung eines Jongleurs im Zirkus [SWOBODA, 2004].

Im Experiment von Chase und Ericsson (1982) wird die Pufferfunktion des Kurzzeitgedächtnisses erläutert. Im Kurzzeitgedächtnis werden große Informationsstücke in chronologischer Reihenfolge zusammengebunden und Informationen wieder abgerufen. Ein weiteres Beispiel ist das Experiment von Baddeley und Hitch (1974). Dabei sollten gleichzeitig Informationen im Gedächtnis gespeichert und Probleme gelöst werden. Im Experiment wurde diese Doppeltätigkeit über das Memorieren eine Liste von unsinnigen Silben und das Lösen von Problemen in Hinsicht auf sinnvolles Denken eingebracht. Die Autoren variierten die Länge der Liste mit unsinnigen Silben, sowie die Schwierigkeit der zu lösenden Probleme. Im allgemeinen ist das Arbeitsgedächtnis zuständig für schlussfolgerndes Denken mit Hilfe eines Zwischenspeichers.

Das Arbeitsgedächtnis ist Teil des dorsolateralen präfrontalen Cortex und liegt im vorderen Teil des Stirnlappens. Dieser Teil ist verbunden mit allen sensorischen (ausgenommen das Geruchssystem), motorischen und limbischen Funktionseinheiten und steuert viele Cortico-Cortico-Bahnen. Es gibt auch eine ausgeprägte Verbindung zum Temporallappen. Diese ist dabei unerlässlich für den aktiven Abruf des Gedächtnisses. Eine Zerstörung des präfrontalen Cortex führt in leichter Form zu Antriebsmangel und kann sich im weiteren Verlauf zu einer Apathie entwickeln. Bei Verletzungen sind normalerweise keine alten Erinnerungen betroffen, jedoch wird das schlussfolgernde Denken beeinträchtigt und Zusammenhänge zwischen Informationen werden schwerer erkannt [GRUBER, 2003/2004].

Das Multikomponenten-Modell des Arbeitsgedächtnisses von Baddely und Hitch wurde als ein theoretisches Bezugssystem vorgestellt, dessen Funktion eine ökonomische und flüssige Darstellung einer großen Breite an Daten geben soll. Es wurde entwickelt, damit die Größe der Kapazität zur Bewerkstelligung verschiedener Aufgaben beurteilt werden kann, sowie als eine Hilfe für die schnellere Beantwortung von Fragen in Hinblick auf das Grundverständnis der Kognition.

Baddeley und Hitch [BADDELEY, 1974] entwickelten das frühere Konzept in ein Drei-Komponenten-System weiter. Dieses Modell beinhaltet die zentrale Exekutive, als eine übergeordnete Kontrollinstanz mit begrenztem Leistungsvermögen, und zwei unterstützende Speichersysteme. Das erste Speichersystem, die phonologische Schleife, betrifft akustische und sprachliche Informationen. Das zweite Teilsystem, der visuell-räumliche Notizblock, führt eine einfache Funktion für visuelle und räumliche Informationen aus (Abbildung 3.1a).

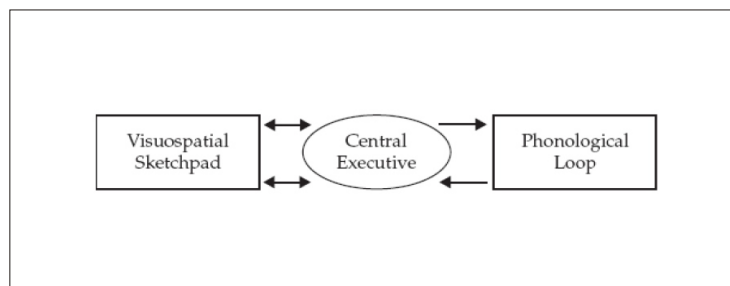


Abbildung 3.1a: Das Working-Memory-Modell entwickelt von Baddeley & Hitch [BADDELEY, 1974] beinhaltet die zentrale Exekutive als übergeordnetes Kontrollsystem und zwei Speichersysteme, den visuell-räumlichen Notizblock und die phonologische Schleife.

Das Multikomponenten-Modell des Arbeitsgedächtnisses hilft den Ablauf von zeitlich gespeicherten Informationen und die Ausführung komplexer kognitiver Prozesse zu verstehen. Der Schwerpunkt des Modells liegt eher auf den Strukturen als auf den Prozessen. Es hat jedoch über Jahre hinweg zu einer Basis für die systematische Ansammlung an Wissen über wichtige kognitive Funktionen geführt [BADDELEY, 2002].

Ein direkter Vergleich von phonologischem und visuell-räumlichem Arbeitsgedächtnis lässt erkennen, dass das visuell-räumliche Arbeitsgedächtnis primär in der rechten Hemisphäre lokalisiert ist. Andere Gebiete, welche allgemein analog zur linken Hemisphärenaktivierung sind, wurden als das sprachliche Arbeitsgedächtnis definiert [BADDELEY, 2003].

Das Arbeitsgedächtnis bezieht sich auf die zeitliche Speicherung und Verarbeitung wichtiger Informationen. Dafür enthält das Arbeitsgedächtnis bereichs-spezifische Kurzzeitgedächtnisspeicher und exekutive Prozesse, wie zum Beispiel Prozesse der Koordination, welche auf dem Inhalt des Arbeitsgedächtnisses arbeiten.

Die Prozesse, bei denen Information zeitlich im Gedächtnis memoriert wird, werden dem Arbeitsgedächtnis zugeordnet. Das Arbeitsgedächtnissystem beinhaltet sprachliche und räumliche Kurzzeitspeicher und exekutive Prozesse, wie multiple Aufgabenbewältigung, Korrektur von Auflösungsstörungen und Gedächtniserneuerung. Diese sind notwendig für hochrangige Denkprozesse und werden vom präfrontalen Cortex unterstützt [BUNGE, 2000].

Die Bedeutung des präfrontalen Cortex für das Arbeitsgedächtnis spiegelt sich in vielen Studien wieder. Zum Beispiel sind Affen mit einer Läsion des frontalen Cortex nicht in der Lage verschiedene Aufgaben zu lösen. Auch Kinder sind bis zum Alter von einem Jahr dazu nicht in der Lage, da in dieser Zeit die Ausreifung des frontalen Cortex stattfindet. Aus Studien von Goldmann an Affen geht hervor, dass unterschiedliche Regionen des präfrontalen Cortex an der Bearbeitung von Informationen beteiligt sind (Abbildung 3.1b). Er vermutet, dass der frontale Cortex in viele kleine Regionen geteilt ist und jede für das Erinnern einer anderen Art von Information zuständig ist. Dies belegen Untersuchungen beim Menschen. Somit scheinen unterschiedliche Areale des Cortex dafür verantwortlich zu sein, unterschiedliche Arten von Informationen im Arbeitsgedächtnis aufrechtzuerhalten [ZENKOVA/RIEGER, 2004/2005]. Der vordere Cortex ist dabei auch beim räumlichen Arbeitsgedächtnis aktiv.

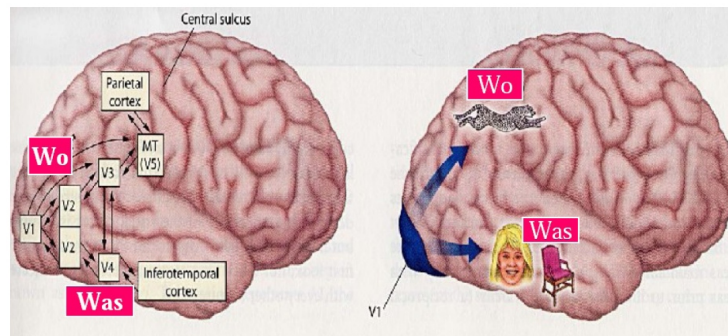


Abbildung 3.1b: Die "WO" und "WAS" Ströme der visuellen Verarbeitung im Cortex [MENZEL]. Die Schwelt ist im Gehirn viele Male abgebildet. Diese Karten sind für jeweils verschiedene Aspekte der Schwelt zuständig.

3.1.1 Die zentrale Exekutive (The central executive)

Die zentrale Exekutive (CE) ist die wichtigste und auch die am wenigsten verstandene Komponente des Arbeitsgedächtnisses. Im ursprünglichen Modell wurde sie als ein Auffangbecken angesehen, in welches alle komplexen Information, die nicht direkt oder spezifischer Weise den beiden Teilsystemen zugeordnet werden konnten, einfloßen [BADDELEY, 2003].

Die zentrale Exekutive ist in ihrer Kapazität beschränkt und wird als ein ausführendes Organ des Arbeitsgedächtnisses angesehen. Sie ist ein attentionales System, welches den visuell-räumlichen Notizblock und die phonologische Schleife überwacht und koordiniert. Die Funktionen der zentralen Exekutive, in Bezug auf Baddeley [BADDELEY, 1996, 2001], sind das Einspeisen von Informationen in jedes der beiden Teilsysteme, sowie das Abrufen daraus und das Übersetzen von Information. Zusätzlich hat die zentrale Exekutive einen Einfluss auf das Zukommen von Aufmerksamkeit für eine Information [ZENKOVA/RIEGER, 2004/2005].

Zu anfangs wurde sie auch als Verbindung zwischen dem visuell-räumlichen Notizblock und der phonologischen Schleife charakterisiert. Bildlich beschrieben arbeitet die zentrale Exekutive als ein

Homunkulus, ein kleiner Mann, welcher die wichtige Entscheidung über die Verwendung der beiden Teilsysteme trifft. Für diese Entscheidung ist ein eigener Übergangsspeicher zuständig. Somit spielt die zentrale Exekutive eine wichtige Rolle im Arbeitsgedächtnis-Bezugssystem.

Frühere Arbeiten konzentrierten sich mehr auf die beiden Teilsysteme, da sie mehr an der Lösung von Problemen beteiligt zu sein schienen als die zentrale Exekutive [BADDELEY, 2002].

Bis heute ist noch nicht geklärt, ob die zentrale Exekutive als "ein einzelnes gleichrangiges System, welches mehrere Funktionen ausführt, oder als eine echte Exekutive, ein großer autonomer Kontrollprozess" betrachtet werden soll [BADDELEY, 2001].

Manche Wissenschaftler sehen in Baddeley's Modell einige Schwierigkeiten. Erstens wird angenommen, dass die zentrale Exekutive in der Lage ist Informationen aus dem Arbeitsgedächtnis mit Informationen aus dem Langzeitgedächtnis zusammenzuführen. Jedoch gibt es bei den bisher beschriebenen Funktionen der zentralen Exekutive keine genauen Informationen darüber, wie die zentrale Exekutive und das Langzeitgedächtnis miteinander interagieren. Zweitens erklärt das Modell nicht, wie und wo verbale und visuelle Informationen aus den beiden Teilsystemen zusammenfließen. Deshalb schlug vor kurzem Baddeley [BADDELEY, 2000] eine vierte Komponente für das Working Memory Modell vor, den episodischen Puffer [GUAN].

3.1.2 Die phonologische Schleife (The phonological loop)

Vermutlich entwickelte sich die phonologische Schleife um die Spracherwerbung zu unterstützen. Die phonologische Schleife beinhaltet zwei Komponenten, einen phonologischen Speicher und artikulatorische Kontrollprozesse. Informationen im phonologischen Speicher zerfallen innerhalb von ungefähr 2 Sekunden, wenn sie nicht aktualisiert werden. Dies ist ein Prozess, ähnlich der Subvokalisierung³, und abhängig von der zweiten Komponente, den artikulatorischen Kontrollprozessen [BADDELEY, 1974].

Die artikulatorische Komponente steht in Zusammenhang mit dem Wortlängeneffekt⁴, wobei eine sofortige Wiederholung einer direkten Aufgabe die Länge der Information bewahren kann [BADDELEY, 1975], da der Speichereffekt bei steigender Wortlänge von einer bis zu fünf Silben stark abnimmt.

Wahrscheinlich spielt der Prozess subvokaler Artikulation eine wichtige Rolle bei der Aufzeichnung von visuell repräsentierten Informationen innerhalb der phonologischen Schleife. Es wird davon ausgegangen, dass der Prozess einen direkten Zugriff auf den phonologischen Speicher bereitstellt [BADDELEY, 1984; MURRAY, 1968]. Dabei kann verbales Material über eine bestimmte Dauer memoriert werden.

Die Stärke des phonologischen Schleifen-Modells liegt in der Tatsache, dass es eine einfache und verständliche Darstellung einer relativ komplexen Menge an Daten repräsentiert.

Baddeley, Gathercole und Papagno zeigten [BADDELEY, 1998], dass die phonologische Schleife bei Kindern die Erwerbung ihrer Muttersprache unterstützt und eine wichtige Rolle bei der Zweitspracherlernung im Erwachsenenalter spielt [BADDELEY, 2002].

Patienten mit phonologischem Kurzzeitgedächtnisverlust, aber mit keiner allgemeinen Sprachbeeinträchtigung, weisen Verletzungen im Temporoparietalgebiet auf. Werden Wörtern visuell präsentiert, zeigen diese Patienten keinen Wortlängeneffekt.

Somit beeinflusst das Sprachverhalten die phonologische Schleife über artikulatorische Kontrollprozesse, wobei die Speicherkomponente verhältnismäßig sprachabhängig erscheint [BADDELEY, 2003].

³ **Subvokalisierung** bezeichnet den Vorgang, dass der Text beim Lesen innerlich mitgesprochen wird. Die Bedeutung dieses Vorgangs ist umstritten und ungeklärt.

⁴ Als **Wortlängeneffekt** wird die Beobachtung beschrieben, dass die Gedächtnisspanne bei kürzeren Worten (gemessen an der Silbenlänge und Sprechdauer) größer ist als bei längeren Wörtern.

3.1.3 Der visuell-räumlicher Notizblock (The visuospatial sketchpad)

Das System des visuell-räumlichen Notizblocks ist imstande visuelle und räumliche Informationen gleichzeitig zu speichern und zu aktualisieren. Damit spielt es eine wichtige Rolle bei der räumlichen Orientierung und in der Lösung von visuell-räumlichen Problemen. Der Notizblock bildet dabei ein Interface zwischen visuellen und räumlichen Informationen, die von den Sinnen, das heißt von visuell-sensorischen Reizen, oder aus dem Langzeitgedächtnis abgerufen werden. Im Modell zeigt sich die Schwierigkeit, die visuelle und räumliche Komponente zu trennen, da beide Komponenten die gleichen Funktionen aufweisen [BADDELEY, 2002].

Der visuell-räumliche Notizblock ist imstande visuell-räumliche Erinnerungen abzurufen und zu aktualisieren. Es wird vermutet, dass die visuelle und räumliche Information als getrennte Komponenten im System bearbeitet werden [GUAN].

Wie sein verbales Äquivalent ist der Notizblock in der Kapazität beschränkt, üblicherweise mit einer Speicherung von drei bis vier Objekten.

3.1.4 Der episodische Puffer (The episodic buffer)

Die zentrale Exekutive bildet ein Interface zwischen ihren Teilsystemen und dem Langzeitgedächtnis [BADDELEY, 1996]. Diese Fähigkeit wurde im ursprünglichen Modell zuerst ignoriert (Abbildung 3.1.4a).

Obwohl Baddeley erklärt, dass der Notizblock die Grundlage für die visuelle Metaphorik⁵ bildet und die phonologische Schleife für die auditive Informationsverarbeitung zuständig ist, gibt es dafür keinen konkreten Beweis [BADDELEY/LOGIE, 1992].

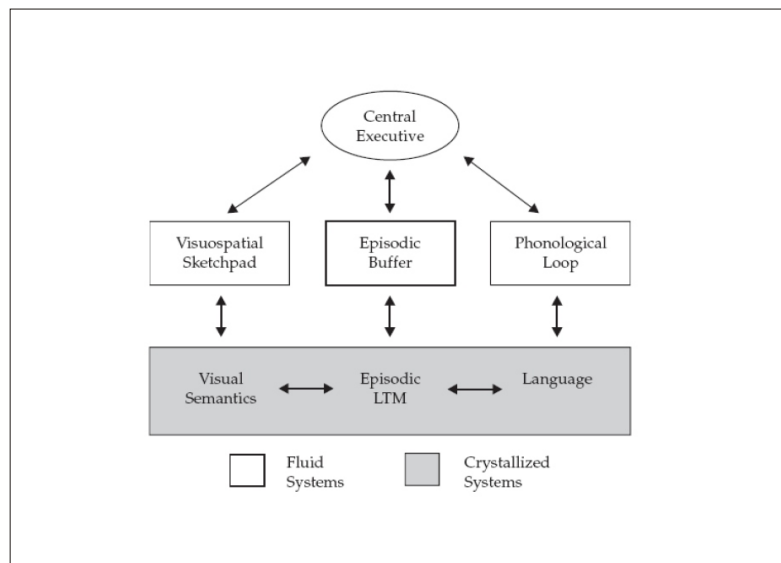


Abbildung 3.1.4a: Das überarbeitete Modell des Arbeitsgedächtnisses, indem Verbindungen mit dem Langzeitgedächtnis (LTM) berücksichtigt werden, in Bezug auf die beiden Teilsysteme und den kürzlich vorgeschlagenen episodischen Puffer [BADDELEY, 2000]. Die dunklen Bereiche zeigen das Langzeitgedächtnis.

⁵ Metaphern spielen bei der Wissensbildung eine wichtige Rolle. Über strukturelle, visuelle und funktionale Analogien können neue Inhalte wesentlich schneller erfasst und verstanden werden.

In Hinsicht auf die im Abschnitt 3.1.1 erwähnten Probleme, soll nun auf die von Baddeley vorgeschlagene Modifikation des Working Memory Modells eingegangen werden, indem ein Blick auf den episodischen Puffer geworfen wird [BADDELEY, A. D., 2000, 2001]. Der episodische Puffer repräsentiert ein Speichersystem, das einen multi-modalen Kodierer verwendet. Es wird angenommen, dass er verschiedene Vorgänge speichern kann und einen Puffer für ein kapazitätbegrenzt Interface zwischen den verschiedenen Teilsystemen bildet, die verschiedene Kodierungen verwenden. Entgegen den Eigenschaften der zentralen Exekutive, deren Funktion weit über das Speichern von Informationen hinausgeht, besitzt der episodische Puffer ausschließlich einen Charakter als Gedächtnisstütze. Es wird angenommen, dass der Puffer die Verbindungsfunktionen bereitstellt, die wichtig für die prinzipiellen biologischen Prozesse der Wahrnehmung sind (Abbildung 3.1.4b). Dies ermöglicht mehrere verschiedene Informationseinheiten gleichzeitig zu betrachten, wobei ein Modell der Umgebung aufgebaut wird, das modifiziert werden kann, um Probleme zu lösen und späteres Verhalten zu planen [JOHNSON-LAIRD, 1983; BADDELEY, 2002].

The working memory system:

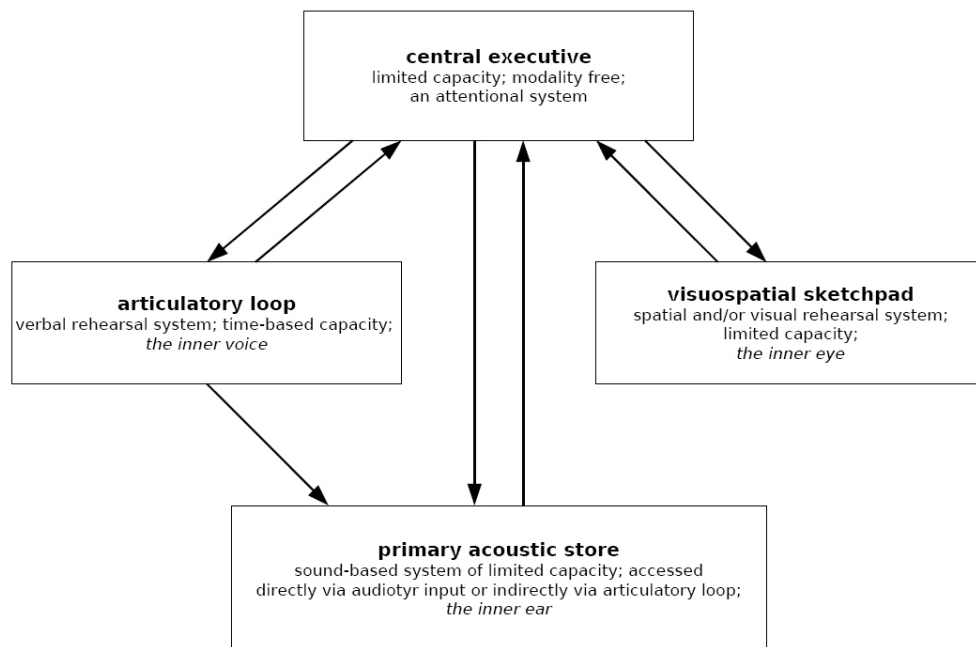


Abbildung 3.1.4b: Das System des Arbeitsgedächtnisses [GUAN].

Der episodische Puffer ist vermutlich der Bereich, an dem Information von den Teilsystemen des Arbeitsgedächtnisses und denen des Langzeitgedächtnisses zusammenfließen. Die Integration von verschiedenen Kodierungen wird wahrscheinlich über die Verwendung einer allgemeinen Kodierung im Puffer gewährleistet. Es wird vermutet, dass der episodische Puffer stark von der zentralen Exekutive abhängt, da es keine direkten Verbindungen zwischen dem Puffer und der phonologischen Schleife, sowie zum visuell-räumlichen Notizblock, gibt. Die Integration der Informationen von beiden Teilsystemen und der Informationen des Langzeitgedächtnisses werden hauptsächlich von der zentralen Exekutive kontrolliert. Anders gesagt, die Rolle des episodischen Puffers ist Arbeitsraum bereitzustellen, indem die Integration von Information, sowie der Einfluss des Langzeitgedächtnisses

auf den Informationsprozess stattfindet. Der episodische Puffer besitzt hierbei die Fähigkeit einer zeitlichen Speicherung von Information mit beschränkter Kapazität. Nach Baddeley's Modell des Arbeitsgedächtnisses wird die Integration der Informationen vom episodischen Puffer abgewickelt, welcher wiederum von der zentralen Exekutiven kontrolliert wird [GUAN].

3.2 Das Langzeitgedächtnis

Definition 13: Das Langzeitgedächtnis ist der dauerhafte Bestand des Wissens und der erworbenen Fertigkeiten einer Person [KÖHLER/MEINEL].

Im Jahr 1983 unterteilte Tulving das Langzeitgedächtnis in ein semantisches und ein Ereignisgedächtnis. Dabei enthalten die Gedächtnisinhalte des semantischen Gedächtnisses keine Raum-Zeitkoordinate. Das Ereignisgedächtnis wird weiter in ein episodisches und ein prospektives Gedächtnis unterteilt. Für das Abrufen aus dem episodischen Gedächtnis ist der vordere Cortex bedeutsam. Des Weiteren wurden von Anderson 1976 zwei Lernarten definiert (Abbildung 3.2). Das deklarative Gedächtnis enthält bewusste, explizit benennbare Konzepte und Fakten, während das prozedurale Gedächtnis über unbewusste, implizite Gedächtnisinhalte verfügt. Das hierzu gehörige Mehrspeichermodell geht auf Cohen und Squire (1989) zurück. Beide Lernarten sind notwendig für die Wissensbildung. Bei einer neuen Aufgabe wird sehr schnell deklaratives Wissen und nur langsam prozedurales Wissen erworben. Jedoch geht das deklarative Wissen im Laufe der Informationsaufnahme auch schneller wieder verloren [SQUIRE, 1992].

Einteilung des Gedächtnisses nach Art des Wissens

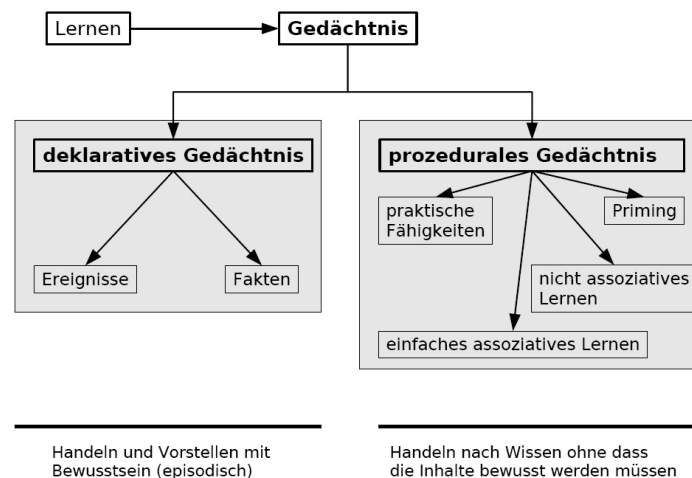


Abbildung 3.2: Einteilung der Gedächtnisse nach der Art des Wissens [MENZEL].

Das Langzeitgedächtnis ist ein Gedächtnisprozess zum Memorieren von Informationen für den Abruf zu einem beliebigen späteren Zeitpunkt. Dabei variieren die verschiedenen Informationsteile des Langzeitgedächtnisses von Zeit zu Zeit in ihrer Verfügbarkeit im Arbeitsgedächtnis. In einer Reihe sogenannter kognitiver Architekturen werden solche Denk- und Problemlöseprozesse simuliert. Eine der bekanntesten ist die ACT-Theorie (Adaptive Control of Thought) von Anderson. Sie besagt, dass sich die Gedächtnisspur über die Darbietung assoziierter Konzepte aktiviert.

Die Bildung der Repräsentation erfolgt semantisch und die Höhe der Aktivierung bestimmt die Wahrscheinlichkeit und die Geschwindigkeit des Zugriffs. Dabei spielen zwei Faktoren für die Aktivierung eine wichtige Rolle. Zum einen, wie kurz der Abruf zurückliegt und zum anderen die Stärke des Übungsgrades. Dabei wird die Annahme unterstützt, dass sich die Aktivierung entlang der Pfade eines Netzwerkes ausbreitet und eine unbewusste Bahnung von Wissenstrukturen erfolgt. Wird eine Information dargeboten, so breitet sich nach der Netzwerktheorie die Aktivierung von der zugehörigen Information zu weiteren, mit der Information assoziierten Gedächtnisinhalten aus. Die Höhe der Aktivierung einer Spur bestimmt dabei die Zugänglichkeit dieser Gedächtnisinhalte und somit kann das Aktivierungsniveau schnell anwachsen. Jedesmal, wenn eine Gedächtnisspur verwendet wird, wächst die Stärke, die zum Teil über die Aktivierung der Spur bestimmt wird [ZENKOVA/RIEGER, 2004/2005].

Das Vergessen von Wissen wird zum einen über den Zerfall der zuvor ausgeprägten Gedächtnisspur erklärt, zum anderen über die Hemmung der Beeinträchtigung des Gedächtnisinhaltes, der Interferenz. Es wird zwischen proaktiver Interferenz, altes Wissen erschwert den Erwerb von neuem Wissen, und retroaktiver Interferenz, neues Wissen überlagert altes Wissen, unterschieden [KÖHLER/MEINEL].

Abschließend ist zu erwähnen, dass im Langzeitgedächtnis Informationen, Bedeutungen, Kategorien, Regeln usw. organisiert und oft zusammen mit Abrufhilfen gespeichert werden. Dabei wird die Bedeutung, also der Sinn der Informationen gespeichert. Diese Informationen wiederum, können bei der Bearbeitung und Speicherung neuen Materials herangezogen werden (top-down Prozesse⁶).

3.3 Das Ortsgedächtnis

In Hinsicht auf alle Kenntnisse im Bereich des "Gedächtnis", sollte ein Blick auf die Frage geworfen werden, was ein Gedächtnis bei Menschen notwendig macht. Vor tausenden von Jahren mussten Menschen noch nach Nahrung und Wasser suchen. Dies war eine viel schwierigere Aufgabe als in der heutigen Zeit, in der an jeder Ecke Supermärkte zu finden sind. So war es notwendig, sich daran zu erinnern, wo die besten Stellen für Nahrung und Wasser liegen. Demzufolge, steht das Gedächtnis in Bezug mit dem Ort beziehungsweise mit dem Ortsgedächtnis.

Das Ortsgedächtnis in Menschen und Tieren scheint nach wissenschaftlichen Studien sehr ausgeprägt zu sein.

⁶ **Top-down Prozesse** greifen auf bekannte Merkmale zurück und steuern vom Gehirn, das heißt, von höheren Verarbeitungsstufen aus, die Auswahl und Interpretation der Sinnesinformation.

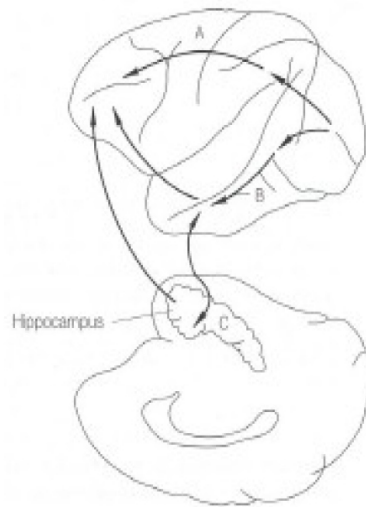


Abbildung 3.3: Die Verbindungen zwischen den einzelnen Cortex-Gebieten sind Grundlage für das Ortsgedächtnis [GRUBER, 2003/2004].

Aus verschiedenen Studien geht hervor, dass sowohl Menschen, als auch Ratten sich einen Ort in einem einzelnen Versuchsdurchgang merken können. Da das Ortsgedächtnis bei Säugern und Vögeln im Hippocampus lokalisiert ist hat man bei verschiedenen Tierstudien Beobachtungen zwischen Hippocampus und räumlicher Orientierung gewonnen (Abbildung 3.3). Vögel, zum Beispiel, können sogar an die tausend verschiedene Orte erkennen, an denen sie Nahrung versteckt haben. Damit gibt es einen Zusammenhang zwischen der Größe des Hippocampus und der räumlichen Orientierung. Dies bestätigt sich darin, dass Vögel, die mehr Nahrungsdepots anlegen können, einen größeren Hippocampus aufweisen [GRUBER, 2003/2004].

Beim Menschen hat der rechte Hippocampus die Fähigkeit das Gedächtnis mit Hilfe der räumlichen Wahrnehmung zu organisieren. Somit korreliert die Genauigkeit der Orientierung mit der Aktivierung des rechten Hippocampus. Ein weiteres Beispiel hierzu findet man unter Londoner Taxifahrern. In einer Studie wurde festgestellt, dass Taxifahrer in London einen größeren hinteren Hippocampus besitzen, und je länger ein Taxifahrer dieser Arbeit nachging, desto größer war sein Hippocampus. Daraus lässt sich schließen, dass der Hippocampus für das Gehirn einen Gedächtnisorganisator darstellt, dessen Organisationsprinzip von Raum und Zeit abhängig ist und für Wegeplanung zuständig sein kann. Da Informationen an vielen Stellen im Gehirn verteilt werden, sorgt der Hippocampus dafür, dass sie an den richtigen Stellen abgelegt und wieder aufgerufen werden können [MAGUIRE, 2000].

3.4 Lernen

In diesem Abschnitt soll kurz erläutert werden, was die Gedächtnisleistung bestimmt. Die Gedächtnisleistung wird nicht nur vom Umfang des Lernens bestimmt, sondern auch von der Verarbeitung des Materials. Eine Verarbeitung mit einem starken bedeutungsbezogenen Inhalt, der Information wird Aufmerksamkeit gewidmet, verbessert die Gedächtnisleistung. Eine elaborierte Verarbeitung, das heißt eine Anreicherung des Materials um zusätzliche Informationen, scheint das Behalten der Inhalte zu verbessern. Ein starker Einfluss geht von inzidentellem und intentionalem Lernen

aus. Es ist entscheidend, wie man das Material lernt (inzidentuell), und nicht so sehr, ob man beabsichtigt das Material zu lernen (intentional).

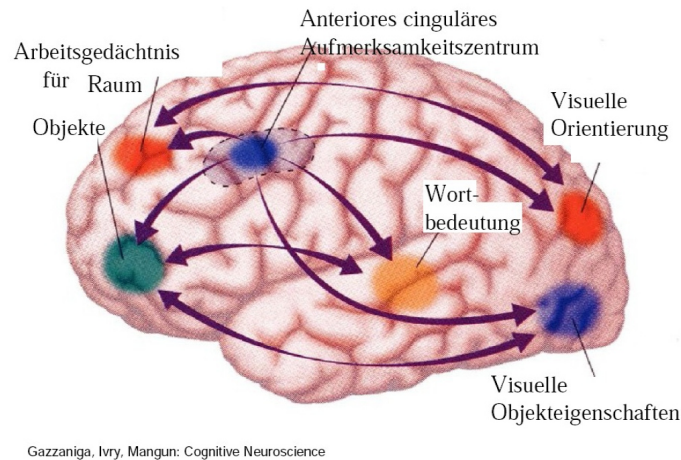


Abbildung 3.4: Nur wenn Aufmerksamkeit auf ein Objekt gerichtet wird, dann wird es gelernt [MENZEL].

Eine wichtige Rolle beim Lernen spielen neuronale Korrelate. Eine stärkere neuronale Verarbeitung führt zu besserem Erinnern (Abbildung 3.4). Wie im vorigen Abschnitt schon erwähnt, ist der Hippocampus ein wichtiger Teil des Gedächtnisses und notwendig für das Speichern von Gedächtnisinhalten. Er stellt einen Präindikator für ein gutes Langzeitgedächtnis dar [ZENKOVA/RIEGER, 2004/2005].

3.5 Eine Studie im Vorfeld

Im Vorfeld der in dieser Arbeit besprochenen Experimente, wurde eine Studie über menschliches Verhalten bei der Lösung von Pfadplanungsaufgaben mit Blick auf die Verwendung einer Nearest Neighbour Strategie untersucht. Hierzu wurden 25 Symbole in verschiedenen Farben auf einem 5 qm großen Gitter zufällig angeordnet. Die Studie sollte zeigen, ob Versuchspersonen die Nearest Neighbour Strategie verwenden. Die Lösung einer TSP-Aufgabe mittels der Nearest Neighbour Strategie beruht auf der Wahl des am nächstgelegenen Punktes ausgehend vom aktuellen Ausgangspunkt (siehe Kapitel 8, Abschnitt 8.1). Für diese Studie wurden Pfade konstruiert, die unter der Verwendung der Nearest Neighbour Strategie zu einem optimalen Pfad führen, sowie Pfade die unter der Verwendung dieser Strategie zu suboptimalen Pfaden führen. In Pfaden, die zu einer suboptimalen Lösung führen, verleitet der nächstgelegene Punkt ausgehend vom aktuellen Ausgangspunkt die Versuchsperson auf einen Umweg. Das Ergebnis der Studie war überraschend, da die Versuchspersonen keine Nearest Neighbour Strategie verwendeten. Bei der Befragung am Ende des Experiments gaben die Versuchspersonen an, dass sie die experimentelle Umgebung in räumliche Regionen unterteilt und anhand dieser Regionen den Pfad geplant haben.

Aufgrund dieser Aussagen wurde das erste Experiment entwickelt um herauszufinden, ob die Versuchspersonen wirklich eine regionen-basierte Planungsheuristik zur Lösung der TSP-Aufgaben verwenden. Solch eine regionen-basierte Heuristik, bildet bei der Pfadplanung einen hierarchischen Prozess. Zuerst wird ein grober Plan generiert in Bezug auf die Verteilung der Regionen. Dieser Plan wird während der Navigation verfeinert. Die Verwendung eines solchen regionen-basierten Planungsschemas untermauert die Hypothese, dass die Versuchspersonen zuerst alle Ziele in einer

Region ablaufen, bevor sie sich in die nächste Region begeben. Es stellt sich in Hinblick auf eine hierarchische Planungsstrategie die Frage, ob die Verwendung einer solchen Strategie dazu führt, den kognitiven Aufwand zu verringern. Um diese Hypothese zu testen, wurde die Umgebung in den beiden folgenden Experimenten in verschiedene Regionen unterteilt.

4 Methoden und Materialien

In diesem Kapitel sollen die Methoden zur Ausführung der Experimente behandelt werden. Es wird dabei ein Blick auf den experimentellen Aufbau geworfen, sowie auf die Aufgabenstellung, die den Versuchspersonen erteilt wurde. Des Weiteren wird besprochen, wie die Auswertung der Daten erfolgte.

4.1 Der experimentelle Aufbau

Die Experimente wurden in einem großen Raum durchgeführt. 25 kleine Pfeiler wurden auf einem 5 qm großen Gitter mit einer Gitterweite von 1,1 m aufgestellt. Unter Beachtung der Lage der Regionen, das heißt, jede Region enthält Symbole der gleichen Farbe, wurden die Symbole zufällig auf den Pfeilern verteilt. Dabei enthalten die Regionen A bis D jeweils 6 verschiedene Symbole der gleichen Farbe und Region E, das 25. Symbol, ein schwarzes Quadrat (siehe Abbildung 4.1a und 4.1b).

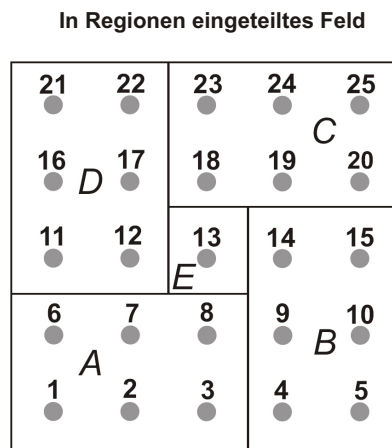


Abbildung 4.1a: Der Aufbau des Feldes mit den 5 Regionen (A - E).

Um Einflüsse auf die Ergebnisse zu kontrollieren, wurden zwei Versionen des Aufbaus generiert, welche sich in der Lage der Regionen und der Anordnung der Symbole in den Regionen A bis D unterscheiden. Die Hälfte der Versuchspersonen führte das Experiment in der ersten Konfiguration aus, die andere Hälfte in der zweiten Konfiguration.

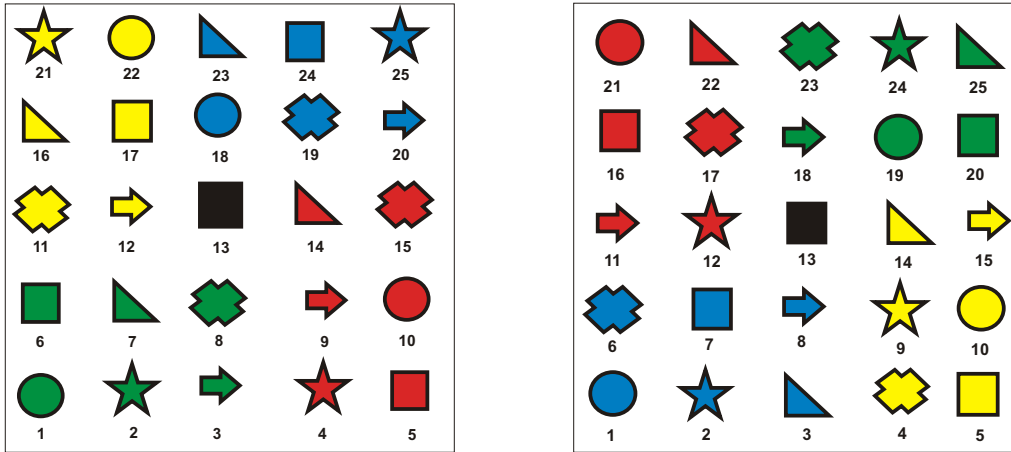


Abbildung 4.1b: Der Aufbau der beiden Feldvariationen mit den farblich eingeteilten Regionen.
 links: Der Aufbau von Feld 1.
 rechts: Der Aufbau von Feld 2.

4.2 Die Navigationsaufgabe

Während des Experiments wurden die Versuchspersonen angewiesen verschiedene TSP-Aufgaben zu lösen. Für jede Aufgabe erhielten sie eine *shopping* Liste (Abbildung 4.2), welche die Symbole enthielt, die den Startpunkt, sowie die Zielpunkte definieren. Diese Zielpunkte sollten innerhalb des Pfades besucht werden. Den Versuchspersonen wurde die Liste mit Blick auf deren Rückseite gegeben, so dass sie zuerst nicht sehen konnten, welche Symbole sich auf der Liste befinden. Es wurde ihnen vom Experimentator mitgeteilt, welches Symbol den Startpunkt definiert, und zu diesem Punkt im Gitter sollten sie sich begeben. Nachdem die Versuchsperson den Startpunkt erreicht hatte, durfte sie die *shopping* Liste umdrehen und beginnen die gestellte Aufgabe zu lösen. Dabei sollte der kürzeste Pfad, ausgehend vom Startpunkt über alle Zielpunkte und zum Startpunkt zurückkehrend, gefunden werden. Jeder Zielpunkt durfte nur ein einziges Mal besucht werden. Während der Navigation markierten die Versuchspersonen die Zielpunkte mit kleinen schwarzen Markierungen.

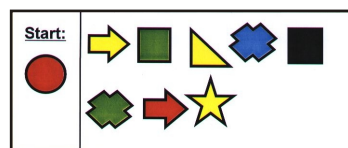


Abbildung 4.2: Beispiel einer Einkaufsliste, welche den Versuchspersonen gezeigt wurden. Auf der rechten Seite ist der gesonderte Startpunkt abgebildet und auf der linken Hälfte befinden sich die Symbole, welche die Versuchspersonen genau einmal besuchen sollten und gleichzeitig die Symbole auf ihrem Weg so anordnen, dass sie dem kürzest möglichen Pfad folgten.

Im zweiten Experiment wird nocheinmal die oben gestellte Navigationsaufgabe verwendet. Erschwerend kommt hier jedoch hinzu, dass die Symbole im Feld während der Navigation abgedeckt

werden. Das heißt, den Versuchspersonen wurde zuerst 10 min lang das Feld mit aufgedeckten Symbolen gezeigt. Dabei sollten sie sich die Symbole, Regionen und deren Lage im Feld so gut wie möglich einprägen. Danach wurden die Symbole abgedeckt und ein kurzes Training in zwei experimentellen Phasen durchgeführt. In der ersten Trainingsphase wurde den Versuchspersonen eine *shopping* Liste mit einem Startpunkt und einem Zielpunkt gezeigt. Diese Symbole sollten im Feld gefunden und aufgedeckt werden. In der zweiten experimentellen Phase sollten der Startpunkt und zwei Zielpunkte gefunden werden. Zwischen jeder Trainingsaufgabe wurde das Feld erneut abgedeckt, so dass jede Aufgabe mit vollständig abgedecktem Feld gelöst werden musste. Während jeder Trainingsaufgabe wurde die Fehlerquote, das heißt, die Anzahl der falsch aufgedeckten Symbole, aufgezeichnet. Wurden am Ende des Trainings fünf Aufgaben nacheinander ohne Fehler bewältigt, wurden den Versuchspersonen sofort die eigentlichen Navigationsaufgaben gestellt. Konnte dieses Kriterium von der Versuchsperson nicht erreicht werden, so wurde ihr noch einmal die Zeit von 5 min zur Betrachtung des Feldes mit vollständig aufgedeckten Symbolen zur Verfügung gestellt.

4.2.1 Pfadtypen

Jede Versuchsperson löste 36 TSP-Aufgaben bestehend aus 4, 5, 6, 7, 8 oder 9 Zielpunkten (siehe Tabelle 4.2.1A). Jede Navigationsaufgabe kann einem von zwei Pfadtypen zugeordnet werden, den regionalen Pfaden und den nichtregionalen Pfaden.

1. Regionale Pfade: Wenn Versuchspersonen eine regionenbasierte Planungsstrategie anwenden, zum Beispiel, wenn die Versuchspersonen zuerst alle Zielpunkte in einer Region besuchen, bevor sie zur nächsten Region weitergehen, ist es möglich, mit dieser Strategie den optimalen Pfad zu finden. Das heißt, eine regionenbasierte Strategie unterstützt das Finden des optimalen Pfades (siehe Abbildung 4.2.1). Die regionalen Pfade wurden so konstruiert, damit sie das Finden einer optimalen Lösung unterstützen. Hierbei folgt der Pfad einer regionenbasierten Strategie, indem die Punkte so angeordnet sind, dass die Versuchspersonen den optimalen Pfad finden kann, falls sie eine Region nach der anderen besucht.
2. Nichtregionale Pfade: Die Verwendung einer regionenbasierten Planungsstrategie auf nichtregionale Pfade kann in diesem Fall zu einem suboptimalen Pfad führen (siehe Abbildung 4.2.1). Die nichtregionalen Pfade wurden so konstruiert um das Finden einer optimalen Lösung unter Verwendung einer regionen-basierten Strategie zu verhindern. Hierbei folgt der Pfad keiner regionen-basierten Strategie. Das bedeutet, um eine optimale Lösung zu finden, muss von einer Region zwischenzeitlich in eine andere Region gewechselt werden. Somit führt die Anwendung einer regionen-basierten Strategie zu einem suboptimalen Pfad.

Route type	number of target places	Start place (target places)
region	4	6(9,5,10,19), 22(12,3,13,20), 6(7,13,15,17)
	5	22(12,2,4,5,18), 17(11,13,4,15,19), 25(13,7,4,5,9)
	6	13(22,20,14,9,5,7), 25(13,7,3,4,5,9), 2(17,21,22,23,19,18)
	7	9(8,1,12,11,21,17,18), 11(6,8,14,10,15,25,19), 1(16,12,13,18,24,14,9)
	8	4(9,14,19,23,16,17,13,2), 10(9,8,13,12,11,16,21,19), 20(18,17,21,16,7,13,9,10)
	9	15(9,3,7,13,12,11,16,22,24), 9(4,5,10,14,19,25,24,18,8), 10(4,8,2,1,16,13,18,24,20)
nonregion	4	23(18,14,4,20), 15(12,6,11,22), 5(15,24,14,7)
	5	20(24,14,7,4,10), 10(18,22,11,6,12), 1(3,5,15,20,7)
	6	12(21,19,13,8,4,6), 20(17,21,16,11,7,12), 10(18,22,16,11,6,12)
	7	6(1,3,9,5,10,20,14), 21(6,12,13,8,4,14,19), 20(19,22,13,12,2,8,9)
	8	11(13,14,20,15,4,8,2,1), 25(20,15,14,9,4,3,2,18), 1(2,3,4,10,19,14,8,11)
	9	20(19,24,23,18,14,9,3,4,10), 6(2,8,4,5,20,13,18,22,16), 11(22,24,20,15,14,13,9,3,7)

Tabelle 4.2.1A: In der Tabelle sind alle Pfade aufgeführt. Der Startpunkt gefolgt von den Zielpunkten (in Klammern). Die Nummern entsprechen den Nummern der Plätze in der schematischen Zeichnung der experimentellen Umgebung (siehe Abbildung 8 und 9).

Anmerkung: Die Konfiguration von Startpunkt und Zielpunkten ist immer identisch bei zwei Paaren von Pfaden. Für einen Pfad aus der Gruppe der regionalen Pfade und für einen aus der Gruppe der nichtregionalen Pfade. Pfade der nichtregionalen Gruppe sind in Bezug auf die komplementären Pfade in der regionalen Gruppe auf dem Gitter verschoben, gedreht oder gespiegelt. Jeder Unterschied im Verhalten bei regionalen Pfaden und nichtregionalen Pfaden ist ein Merkmal der Pfadcharakteristik.

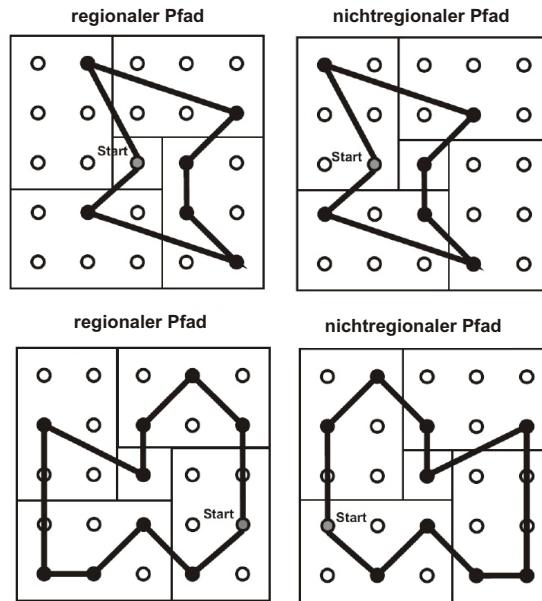


Abbildung 4.2.1: Beispiel für regionale Pfade und nichtregionale Pfade.

oben: Route mit 6 Zielpunkten; eine regionen-basierte Strategie führt links zu einem optimalen Pfad, rechts jedoch zu einem suboptimalen.

unten: Route mit 9 Zielpunkten; eine regionen-basierte Strategie führt links zu einem optimalen Pfad, rechts jedoch zu einem suboptimalen.

4.3 Versuchspersonen

Jeweils 24 Versuchspersonen nahmen an den beiden Experimenten teil. Die meisten waren Studenten der Universität Tübingen mit einem Durchschnittsalter von 26 Jahren. Es wurden 8 Euro pro Stunde bezahlt.

4.4 Auswertung

Die Pfade der Versuchspersonen, das heißt, die Reihenfolge der nacheinander besuchten Plätze, wurde für jede Navigationsaufgabe aufgezeichnet und die Länge des Pfades berechnet. Zum Vergleich wurde der kürzest mögliche Pfad über die Euklidische Distanz für jede dieser Aufgaben zuvor berechnet. Die Fehlerquote (*Overshoot*) wurde mit Hilfe der Formel (4.4) berechnet.

$$Overshoot = \left(\frac{Laenge\ Pfad\ der\ Versuchsperson}{Laenge\ des\ optimalen\ Pfades} - 1 \right) \cdot 100 \quad (4.4)$$

Eine Fehlerquote von 100 % entspricht einem Pfad mit doppelter Länge des optimalen Pfades. Im Weiteren wurde der Prozentsatz für die Aufgaben berechnet, in denen die Versuchspersonen den optimalen Pfad gefunden haben. Für jede Aufgabe wurde auch die vergehende Zeit bis zum initiieren der Aufgabe gemessen, das heißt die Zeit, die verging vom Umdrehen der *shopping* Liste bis zum Beginn der Navigation.

Des Weiteren wurden die Pfade der Versuchspersonen auf regionaler Ebene analysiert. Dafür wurde die Anzahl der Überschreitungen von Regionen jedes Pfades einer Versuchsperson und der optimalen Pfade berechnet. Das bedeutet, um eine Aufgabe zu lösen, muss zwischen den Regionen gewechselt werden. Wie häufig die Regionen gewechselt werden müssen, kann anhand der optimalen Lösung berechnet werden. Wie oft eine Versuchsperson die Region gewechselt hat, wird an der aufgezeichneten Reihenfolge der zu besuchenden Zielpunkte berechnet. Beim Vergleich von optimaler Anzahl an Regionenüberschreitungen und der Anzahl an Überschreitungen, die eine Versuchsperson während der Navigation verursacht hat, ist zu erkennen, ob eine regionen-basierte Strategie verwendet wurde.

Hinzukommend wurde jeder Punkt eines Pfades auf regionaler Ebene definiert. Beispielsweise wird ein Pfad über den Platz seiner Symbole beschrieben, indem die Plätze des Startpunktes und der Zielpunkte eine Nummer zugewiesen bekommen (hier: 6 (9, 5, 10, 19)), so bekommen auf regionaler Ebene die Plätze eine Region zugewiesen (hier: A (B, B, B, C)). Beim Vergleich der Regionendarstellung eines Pfades der Versuchsperson mit der Regionendarstellung des entsprechenden optimalen Pfades, können die Fehler basierend auf regionaler Ebene unabhängig von den Fehlern auf Ebene der Plätze betrachtet werden.

Die Auswertung erfolgt im ersten Experiment auf Basis der korrekt besuchten Zielpunkte, das heißt, besitzt ein Pfad einen beziehungsweise mehrere falsche Zielpunkte, die im vorgegebenen Pfad nicht vorkommen, fehlte ein vorgegebener Zielpunkt oder waren zu viele Zielpunkte vorhanden, wurden diese Pfade aus der Auswertung herausgenommen, da entweder ein Fehler des Experimentators oder der Versuchsperson selbst vorlagen. Es wurden nur falsch aufgezeichnete Pfade entfernt, die Startzeiten gingen jedoch mit in die Auswertung mit ein. Im zweiten Experiment wurde genauso verfahren, es werden hier jedoch zwei getrennte Auswertungen vorgenommen. Einmal eine Auswertung nur mit den Pfaden, die die korrekten Zielpunkte enthalten und des Weiteren eine Auswertung, in die Pfade mit den richtigen Zielpunkten plus den falsch aufgedeckten Symbolen mit einbezogen wurden, da diese Pfade Auskunft über die Rolle des Gedächtnisses geben können. Pfade bei denen Fehler des Experimentators oder der Versuchspersonen vorlagen, fließen wiederum nicht in die Auswertung mit ein. Auch hier werden nur die Pfade aus der Auswertung genommen, nicht die Startzeiten.

Die Auswertung der Daten erfolgte in R, einer Sprache und Umgebung für statistische Berechnungen und Graphiken. R ist ein GNU Projekt, welches der Sprache S ähnelt und eine Umgebung besitzt, die von John Chambers und Kollegen der Bell Laboratories (zuerst AT&T, jetzt Lucent Technologies) entwickelt wurde. R ist ein Open Source Projekt und frei im Internet erhältlich. R kann als eine etwas andere Implementation als S angesehen werden. Daher gibt es einige wichtige Unterschiede, jedoch funktioniert meist der in S geschriebene Code auch unter R.

R unterstützt eine große Vielfalt an statistischen und graphischen Methoden, und ist erweiterbar. Die Sprache S ist meist das gewählte Auswertungswerkzeug für die Wissenschaft in statistischen Methoden. R unterstützt jedoch wissenschaftliche Auswertungen, auch in Bezug auf die Sprache S.

R ist eine freie Software und kann unter der Free Software Foundation's GNU General Public License erhalten werden. Es kann unter verschiedenen Systemen (LINUX, verschiedene UNIX-Plattformen, Windows und MacOS) installiert werden.

R ist eine reine Computersprache und erlaubt dem Benutzer neue Funktionen zu implementieren. Der Code vieler erhältlicher Funktionen ist meist in R geschrieben, jedoch kann Code in C, C++ und Fortran integriert werden. R kann leicht mit verschiedenen Packages erweitert werden. Acht dieser Packages sind im R-Installationspaket enthalten, viele weitere sind über die CRAN Internetseiten erhältlich.

Für die Auswertung der Daten wurden Auswertungsprogramme für die Fehlerquote, für die Berechnung der Pfadlänge, der Startzeit usw. implementiert, sowie für Berechnungen im Bereich der regionen-basierten Strategie und der Trainingsdaten des zweiten Experiments.

5 Ergebnisse Experiment 1

In diesem Kapitel werden die Ergebnisse dargestellt, die aus dem ersten Experiment erhalten wurden. Dabei wird zum einen ein Blick auf die gemessene Startzeit geworfen, die Versuchspersonen benötigten, zwischen Betrachten der *shopping* Liste und beginnender Navigation. Der Hauptteil der Auswertung der Ergebnisse beruhte jedoch auf den produzierten Fehlern, dem *Overshoot*, und der Anzahl der Pfade, die mit der optimalen Lösung übereinstimmten.

Wie in Abschnitt 3.5 beschrieben wurde, soll hier untersucht werden, ob die Versuchspersonen eine regionen-basierte Strategie verwenden um den kognitiven Aufwand zu verringern. Unter dieser Annahme, das heißt, zuerst die Planung eines groben Pfades auf Regionenebene und eine Verfeinerung des Pfades während der Navigation, wurde im Falle von nichtregionalen Pfaden verhindert, dass die Versuchspersonen den optimalen Pfad finden konnte (Abbildung 5).

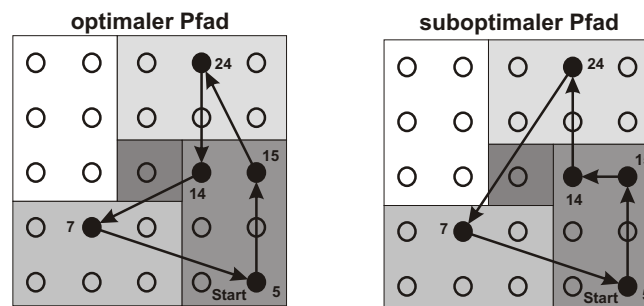


Abbildung 5: Skizzierte Abbildung der experimentellen Umgebung von Experiment 1 mit der Darstellung einer Pfades, einmal als optimale Lösung und einmal mit einer suboptimalen Lösung resultierend aus der regionen-basierten Strategie.

Deshalb wird angenommen, dass die Versuchspersonen höhere Fehlerquoten bei nichtregionalen Pfaden aufweisen, als bei den komplementären regionalen Pfaden. Dabei sollte diese Strategie bei nichtregionalen Pfaden zu einer geringeren Anzahl an Überschreitungen der Regionengrenzen führen als beim optimalen Pfad vorgesehen ist. Die regionalen Pfade werden im folgenden nun in eine regionale Pfadgruppe und die nichtregionalen Pfade in eine nichtregionale Pfadgruppe unterteilt.

5.1 Startzeit

Die Startzeit ist die Zeit, die die Versuchsperson vom Betrachten der *shopping* Liste bis zum Beginn der Navigation benötigte. Die durchschnittliche Startzeit lag bei 16,17 Sekunden. Die Startzeit zwischen der regionalen Pfadgruppe und der nichtregionalen Pfadgruppe unterschied sich in nur geringer Weise. Die durchschnittliche Startzeit bei regionalen Pfaden betrug 16,38 Sekunden und die bei nichtregionalen Pfaden 15,95 Sekunden. Eine ANOVA⁷ zeigte hierbei keinen signifikanten Effekt ($F(1,23) = 0,907$; $p = 0,351$; $\eta^2 = 0,038$; MSE ⁸ = 2,249). Somit schien der Pfadtyp

⁷ ANOVA: *Analysis of Variance*; beschreibt die Vergleichsmöglichkeiten verschiedener Stichproben.

⁸ MSE: mean square of estimation

keinen Einfluss auf die Startzeit zu haben.

Beim Blick auf die Startzeit in Abhängigkeit der Anzahl der Zielpunkte trat ein Anstieg der Startzeit mit Anstieg der Anzahl an Zielpunkten auf (Pearson's product-moment correlation: $p < 2 \cdot 10^{-16}$, $r = 0,81$) (Abbildung 5.1). Eine ANOVA zeigte hier einen starken signifikanten Effekt ($F(5,115) = 5,378$; $p = 0,012$; $\eta^2 = 0,190$; $MSE = 1775,586$). Es zeigte sich, dass die Zielpunkte einfluss auf die Startzeit haben.

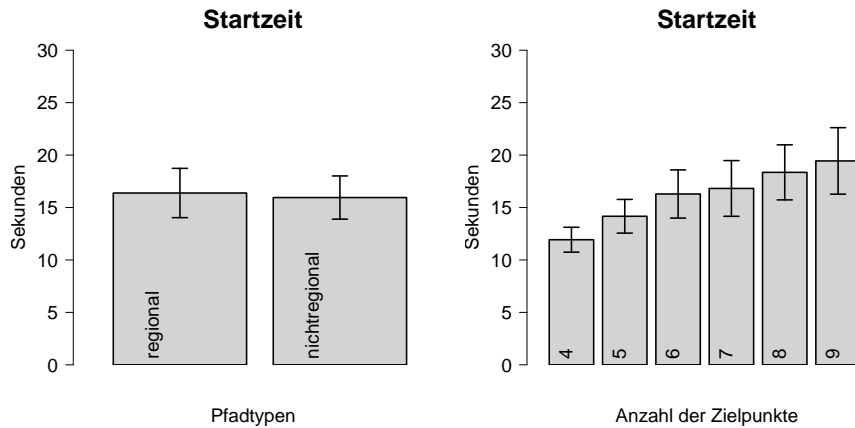


Abbildung 5.1: Startzeiten der Versuchspersonen.

links: Startzeit bei regionalen und nichtregionalen Pfaden.

rechts: Startzeiten nach Anzahl der zu besuchenden Zielpunkte.

Die Startzeit zwischen Männern und Frauen ließ erkennen, dass die Gruppe der Frauen mit einer durchschnittlichen Startzeit von 14,78 Sekunden etwas weniger Zeit zur Planung des Pfades benötigten, als die Gruppe der Männer, die eine durchschnittliche Startzeit von 18,48 Sekunden aufwiesen. Die Startzeit zwischen den regionalen und nichtregionalen Pfaden unterschied sich bei Männern und Frauen kaum. In der Gruppe der Männer trat eine durchschnittliche Startzeit in der regionalen Pfadgruppe von 19,19 Sekunden und von 17,78 Sekunden in der nichtregionalen Pfadgruppe auf. Frauen benötigten in der regionalen Pfadgruppe im Schnitt 14,70 Sekunden und in der nichtregionalen Pfadgruppe 14,86 Sekunden.

5.2 Fehlerquote (*Overshoot*)

Die Fehlerquote berechnete sich mittels der Formel 4.4 im vierten Kapitel. Die durchschnittliche Fehlerquote betrug 5,03 %. Hierbei lag die Fehlerquote bei den nichtregionalen Pfaden etwas höher als bei den regionalen Pfaden. Es konnte jedoch kein eindeutiger Hinweis gefunden werden, dass die Versuchspersonen mehr Fehler in den nichtregionalen Pfad-Aufgaben machten als zuvor angenommen (Abbildung 5.2a oben). Eine ANOVA zeigte keinen signifikanten Effekt bei der Anzahl an Zielpunkten ($F(5,115) = 0,956$; $p = 0,426$; $\eta^2 = 0,040$; $MSE = 27,031$), sowie keinen signifikanten Effekt zwischen den Pfadtypen ($F(1,23) = 2,906$; $p = 0,102$; $\eta^2 = 0,112$; $MSE = 6,051$). Somit schienen weder die Pfadtypen, noch die Anzahl der Zielpunkte einen Einfluss auf den *Overshoot* zu haben.

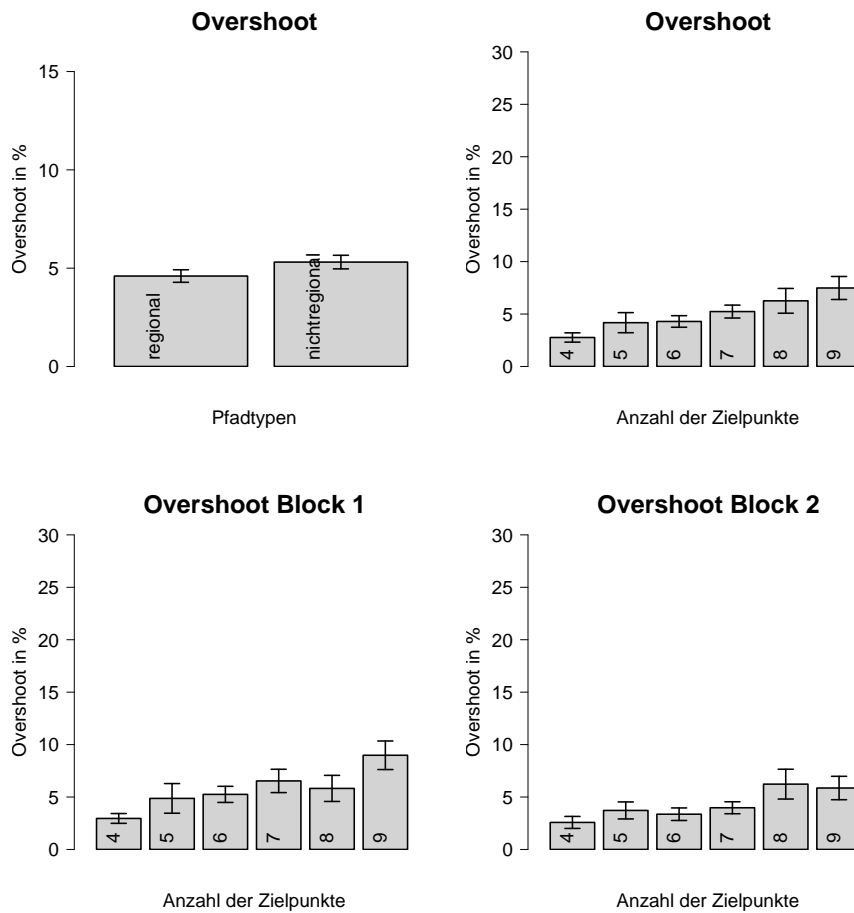


Abbildung 5.2a: Overshoot.

oben: prozentualer *Overshoot* bei regionalen und nichtregionalen Pfaden, und nach Anzahl der Zielpunkte.

unten: *Overshoot* im Vergleich zwischen experimentellem Block 1 und Block 2.

Des Weiteren ließ sich im experimentellen Block 2 eine leichte Verbesserung der Fehlerquote gegenüber dem experimentellen Block 1 erkennen (paired t-test: $t = 3,21$, $df = 23$, $p < 0,004$) (Abbildung 5.2a unten). Jeder experimentelle Block enthielt 18 Pfade mit jeweils gleicher Anzahl an Zielpunkten. Jede Versuchsperson hatte nun entweder zuerst die erste Hälfte der Pfade (Pfade 1 - 18) oder zuerst die zweite Hälfte der Pfade (Pfade 19 - 36) gelöst. Das bedeutet, im ersten experimentellen Block war die Fehlerquote enthalten, die von den 18 Pfaden stammt, die zuerst gelöst wurden, im zweiten experimentellen Block waren dagegen die Pfade enthalten, die in der zweiten Hälfte gelöst wurden. Eine Verbesserung im zweiten experimentellen Block bedeutete deshalb, dass sich eine Verbesserung der Lösung der Aufgaben einstellte, je mehr Aufgaben schon gelöst wurden. Die Einteilung in die beiden experimentellen Blöcke schien somit einen Einfluss auf den *Overshoot* zu haben.

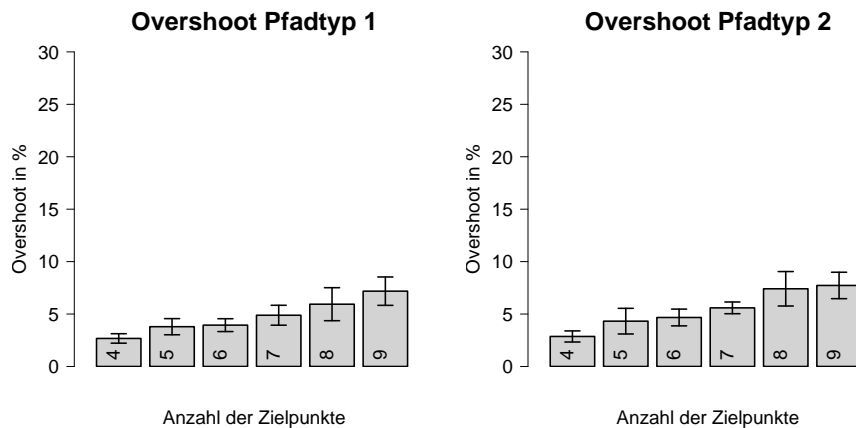


Abbildung 5.2b: *Overshoot* im Vergleich zwischen den beiden Pfadtypen anhand der Anzahl der Zielpunkte.

links: regionale Pfadgruppe.
rechts: nichtregionale Pfadgruppe.

Wie schon oben erwähnt, gab es keinen großen Unterschied in der Fehlerquote zwischen den Pfadtypen. Die durchschnittliche Fehlerquote in der regionalen Pfadgruppe betrug 4,6 %, während sie in der nichtregionalen Pfadgruppe 5,31 % betrug (Abbildung 5.2b). Sie stieg mit höherer Anzahl an Zielpunkten in beiden Fällen leicht an (Pearson's product-moment correlation: $p < 0,0002$, $r = 0,306$).

Die Gruppe der männlichen Versuchspersonen wies einen gering höheren *Overshoot* mit 5,47 % auf, als die Gruppe der weiblichen Versuchspersonen mit 4,76 %. In Hinsicht auf die regionale Pfadgruppe unterschied sich der durchschnittliche *Overshoot* fast nicht. Männliche Versuchspersonen hatten im Durchschnitt einen *Overshoot* von 4,86 %, weibliche einen von 4,56 %. Der Unterschied von fast 1 % kam von der nichtregionalen Pfadgruppe. Hier wurden 6,09 % bei männlichen Versuchspersonen und nur 4,95 % bei weiblichen Versuchspersonen erhalten. Trotz einer längeren Startzeit, wiesen Männer einen etwas höheren *Overshoot* auf als die Frauen.

5.3 Korrekte Pfade

Die durchschnittliche Anzahl an korrekt gefundenen Pfaden lag bei 29,85 % pro Experimentdurchführung. Das heißt, ungefähr 30 % der gelösten TSP-Aufgaben entsprachen der optimalen Lösung, die zuvor berechnet wurde. Dies waren die korrekt gefundenen oder optimalen Pfade. In der regionalen Pfadgruppe wurden mit 38 % durchschnittlich mehr korrekte Pfade gefunden als in der nichtregionalen Pfadgruppe mit 22 % (Abbildung 5.3 links). Eine ANOVA zeigte hier einen signifikanten Effekt ($F(1,23) = 24,602$; $p = 0,000$; $eta^2 = 0,517$; $MSE = 0,292$). Somit hatten die unterschiedlichen Pfadtypen einen Einfluss auf die Anzahl der gefundenen optimalen Pfade.

Beim Blick auf die korrekt gefundenen Pfade in Abhängigkeit von der Anzahl der Zielpunkte ließ sich nur feststellen, dass bei Pfaden mit nur vier Zielpunkten mehr als 40 % der optimalen Pfade von den Versuchspersonen gefunden wurden. Bei den restlichen Zielpunkten ließ sich jedoch keine Tendenz feststellen (Pearson's product-moment correlation: $p = 0,605$, $r = 0,043$) (Abbildung 5.3 rechts). Eine ANOVA zeigte bei den Zielpunkten keinen signifikanten Effekt ($F(5,115) = 0,343$; $p = 0,886$; $eta^2 = 0,015$; $MSE = 0,016$). Die Anzahl der Zielpunkte hatte keinen Einfluss auf die Anzahl der gefundenen optimalen Pfade. Dies steht im Gegensatz zu der im Vorfeld getroffenen Annahme, dass es schwerer werden sollte die optimalen Pfad zu finden, je mehr die Anzahl der Zielpunkte ansteigt.

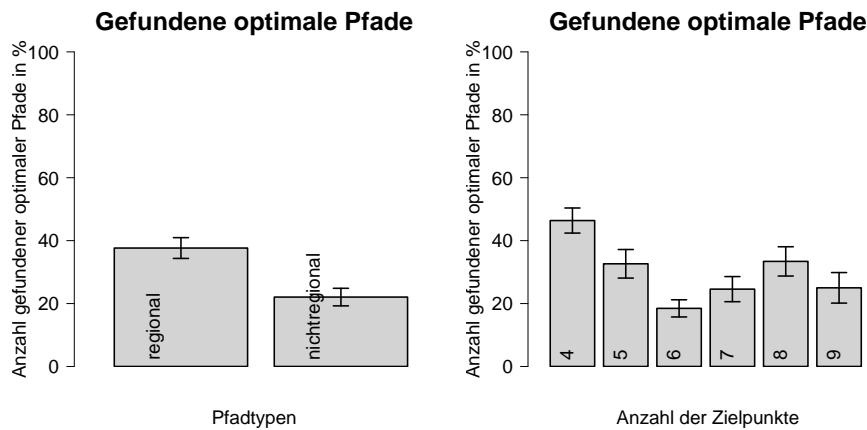


Abbildung 5.3: Anzahl der korrekt gefundenen Pfade. Eimal im Vergleich zwischen den beiden Pfadtypen (links) und einmal nach Anzahl der Zielpunkte (rechts).

Auch ein t-Test zwischen der regionalen und nichtregionalen Pfadgruppe wies eine Signifikanz auf (Welch Two Sample t-test: $t = 3,61$, $df = 44,745$, $p = 0,0008$; paired t-test: $t = 4,96$, $df = 23$, $p = 5,15 \cdot 10^{-5}$). Somit lies sich ein relativ starker Effekt zwischen regionaler und nichtregionaler Pfadgruppe nachweisen.

Die männliche Gruppe der Versuchspersonen wies mit durchschnittlich 32,26 % eine höhere Anzahl an korrekt gefundenen Pfaden auf als die weibliche Gruppe mit 28,54 %. In der regionalen Gruppe der Pfade fanden Männer in 39,96 % der Fälle den optimalen Pfad, Frauen in 36,26 % der Fälle. Eine etwas geringere Anzahl an korrekt gefundenen Pfaden erschien bei nichtregionalen Pfaden. Hier fand die männliche Gruppe der Versuchspersonen in 24,32 % der Fälle und die weibliche Gruppe in 20,70 % der Fälle die optimale Lösung. Dies zeigte, dass trotz einer geringeren Fehlerquote der Frauen, männliche Versuchspersonen dennoch häufiger den optimalen Pfad gefunden hatten.

5.4 Überschreitungen der Regionengrenzen

Im Laufe einer Navigation wurden Regionengrenzen überschritten, das heißt, um die Aufgabe zu lösen, musste eine Versuchsperson die Regionen wechseln. Somit konnte errechnet werden, wie häufig die Versuchsperson die Regionen wechseln musste, falls sie eine optimale Lösung der Aufgabe erhalten wollte. Im Falle von nichtregionalen Pfaden bedeutete dies, dass die Anwendung einer regionen-basierten Strategie zu weniger oder einer gleichen Anzahl an Regionenüberschreitungen führte. Einige der nichtregionalen Pfade hatten die Eigenschaft, dass die Verwendung einer regionen-basierten Strategie zu einer optimalen Anzahl an Regionenüberschreitungen führte, jedoch nicht zu einem optimalen Pfad.

Bei der Lösung von nichtregionalen Pfad-Aufgaben überschritten die Versuchspersonen durchschnittlich 0,65 mal weniger die Regionengrenzen, als dass sie dies bei den optimalen Lösungen tun sollten (siehe Abbildung 5.4 und Abbildung A.1a und A.1b im Anhang A).

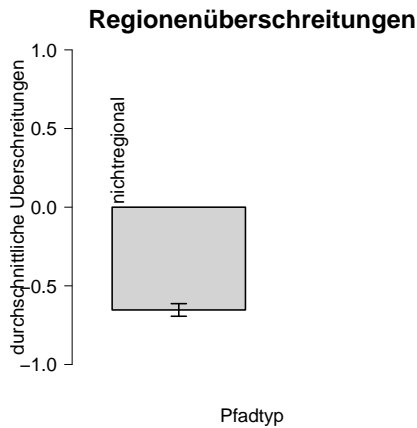


Abbildung 5.4: Durchschnittliche Anzahl der Regionenüberschreitungen bei nichtregionalen Pfaden.

Es trat eine starke Signifikanz zwischen regionaler und nichtregionaler Pfadgruppe auf (Welch Two Sample t-test: $t = 5,49$, $df = 35,58$, $p = 3,43^{-06}$; paired t-test: $t = 5,92$, $df = 23$, $p = 4,93^{-06}$; Pearson's product-moment correlation: $p = 1,2^{-10}$, $r = 0,504$). Auch eine ANOVA zeigte einen signifikanten Effekt zwischen den Pfadtypen ($F(1,23) = 100,764$; $p = 0,000$; $\eta^2 = 0,814$; $MSE = 1,263$). Somit schienen die Pfadtypen einen Einfluss auf die Überschreitung der Regionengrenzen zu haben.

5.5 Regionen-basierte Strategie

Anhand der oben erwähnten Regionenüberschreitungen konnte berechnet werden, wie häufig eine regionen-basierte Strategie verwendet wurde. Dabei wurde zuerst für die optimale Lösung der 36 Pfade berechnet, wie häufig die Versuchsperson die Regionen wechseln musste, falls sie eine regionen-basierte Strategie verwendet hat. Mit regionen-basierter Strategie ist dabei gemeint, dass die Versuchsperson alle Zielpunkte in einer Region besuchte, bevor sie in die nächste Region wechselte. Dann wurde anhand der Pfade der Versuchspersonen die Anzahl an überquerten Regionengrenzen berechnet und mit der Anzahl an Regionenüberschreitungen der optimalen Lösung verglichen. Dies ergab, dass die Versuchspersonen die regionen-basierte Strategie in der regionalen Pfadgruppe durchschnittlich 0,55 mal und in der nichtregionalen Pfadgruppe durchschnittlich 0,23 mal verwendeten (Abbildung 5.5 links). Dieser Unterschied zeigte, dass auch in der nichtregionalen Gruppe die regionen-basierte Strategie verwendet wurde, jedoch nicht so häufig, wie in der Gruppe der regionalen Pfade. Dabei verwendeten männliche Versuchspersonen die regionen-basierte Strategie bei beiden Pfadtypen in etwa so häufig, wie weibliche (siehe Anhang A, Tabelle A.2A).

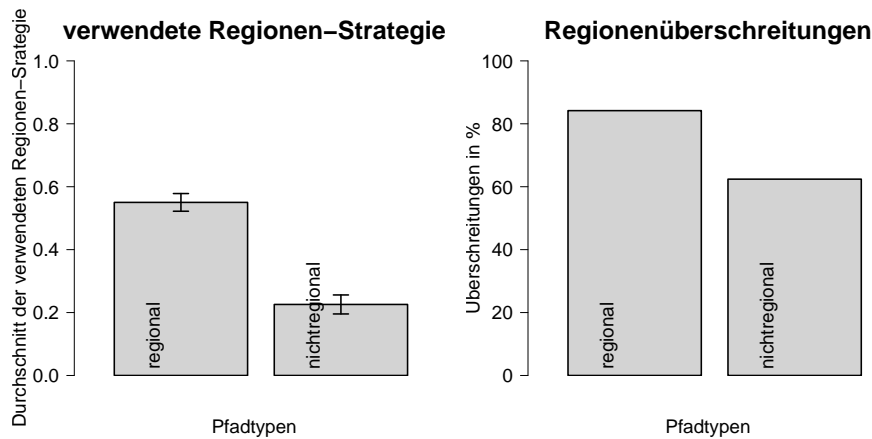


Abbildung 5.5: Regionen-basierte Strategie.
 links: Die im Durchschnitt verwendete Regionen-Strategie.
 rechts: Häufigkeiten der optimalen Regionenüberschreitungen.

84,81 % der Pfade in der regionalen Gruppe und 62,41 % der Pfade in der nichtregionalen Gruppe wurden mit der optimalen Anzahl an Regionenüberschreitungen gelöst (Abbildung 5.5 rechts). Dabei war zu erkennen, dass auch in der Gruppe der nichtregionalen Pfade häufig die optimale Anzahl an Regionenüberschreitungen erreicht wurde. In der nichtregionalen Pfadgruppe lies sich jedoch bei einigen Pfaden nicht unterscheiden, ob eine regionen-basierte Strategie verwendet wurde oder nicht, da bei einigen Pfaden dieser Gruppe die Verwendung einer regionen-basierten Strategie zu der gleichen Anzahl an Regionenüberschreitungen führte, wie diese für die optimalen Regionenüberschreitungen berechnet wurde.

5.6 Beste und schlechteste Pfade

Die besten und schlechtesten Pfade wurden anhand des *Overshoots* eines jeden Pfades von jeder Versuchsperson berechnet. Dabei wurden die Pfade mit einem *Overshoot* kleiner als 1 % in der Gruppe der besten Pfade summiert und Pfade, deren *Overshoot* mehr als 6 % betrug in der Gruppe der schlechtesten Pfade. Somit erhielt man in der Gruppe der besten Pfade, die Pfade 11, 24 und 33 aus der regionalen Pfadgruppe und die Pfade 13 und 35 aus der nichtregionalen Pfadgruppe. Unter die schlechtesten Pfade fielen der Pfad 14 aus der regionalen Pfadgruppe und Pfad 15 aus der nichtregionalen Pfadgruppe.

Durchschnittlich betrug die optimale Pfadlänge aller 36 Pfade eine Länge von 1397,36 cm. Die Pfadlänge der Versuchspersonen betrug im Schnitt 1467,43 cm, darunter die durchschnittliche Länge der regionalen Pfade mit 1460,43 cm und die der nichtregionalen Pfade mit 1474,5 cm. Dieses Ergebnis deutete auf den geringen Unterschied der Fehlerquoten zwischen regionalen und nichtregionalen Pfaden hin.

5.7 Gezeichnete Pfade aller Versuchspersonen

Für jede Lösung der TSP-Aufgabe einer jeden Versuchsperson wurde der Pfad anhand der Reihenfolge der besuchten Zielpunkte rekonstruiert und alle dieser Lösungen übereinander in einer schematischen Abbildung der experimentellen Umgebung aufgezeichnet. In Abbildung 5.7 kann man erkennen, wie häufig die Versuchspersonen eine bestimmte Strecke zurückgelegt hatten. Dies zeigt sich an der Stärke der Linie. Je mehr Versuchspersonen eine Strecke zwischen zwei Punkten

verwendet hatten, desto breiter zeigt sich die Linie in der Abbildung. In der linken Spalte sind die regionalen Pfade dargestellt, in der rechten Spalte die komplementären nichtregionale Pfade.

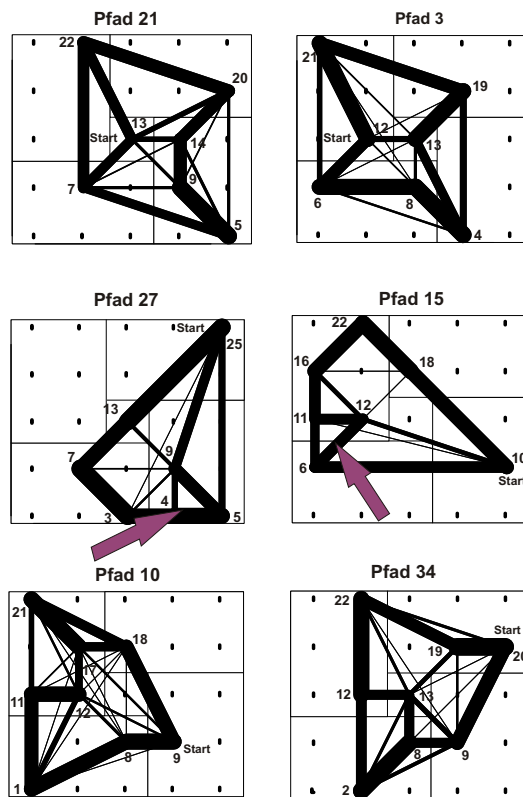


Abbildung 5.7: Pfade der Versuchspersonen (linke Spalte: regionale Pfade; rechte Spalte: nichtregionale Pfade).

In Abbildung 5.7 zeigte sich deutlich, ob eine regionen-basierte Strategie verwendet wurde, und welche Auswirkung dies auf den geplanten Pfad, speziell bei nichtregionalen Pfaden, hatte. In Pfad 27 ist deutlich zu erkennen, dass die meisten Versuchspersonen einer regionen-basierten Strategie folgten und deshalb auch häufig der optimale Pfad gefunden wurde. Im Gegensatz zu Pfad 15, indem eben diese Verwendung der regionen-basierten Strategie zu einem suboptimalen Pfad führte.

5.8 Kontrollfaktoren

Die Kontrollfaktoren hatten keinen erkennbaren Einfluss auf die Fehlerquote, die Startzeit oder das Finden der optimalen Lösung (Tabelle 5.8A und B).

Liste	Liste A	Liste B	t	df	p-Wert
<i>Overshoot</i> in %	Ø 4,83	Ø 5,23	0,32	22	0,75
Startzeit in Sekunden	Ø 16,31	Ø 16,03	0,06	22	0,95
korrekt gefundene Pfade in %	Ø 28,28	Ø 29,02	0,62	22	0,54

Tabelle 5.8A: Kontrollfaktor Feld. Zwei verschiedene Layouts des Feldes (Liste A und B).

Konfiguration	Konfiguration A	Konfiguration B	t	df	p-Wert
<i>Overshoot</i> in %	Ø 5,69	Ø 4,37	1,09	22	0,29
Startzeit in Sekunden	Ø 17,19	Ø 15,15	0,46	22	0,65
korrekt gefundene Pfade in %	Ø 31,74	Ø 28,13	0,68	22	0,50

Tabelle 5.8B: Kontrollfaktor Konfiguration. Zwei verschiedene Layouts der *shopping* Listen (Konfiguration A und B).

5.9 Ausfallpfade

Unter Ausfallpfaden versteht man die Pfade, in denen Fehler, wie falsche Zielpunkte oder das Fehlen von Zielpunkten, auftraten. Da es bei diesen Fehlern teilweise nicht möglich war zurück-zuverfolgen, ob Versuchsperson oder Experimentator den Fehler verursacht hatten, wurden diese Pfade aus der Auswertung herausgenommen. Die Höhe der Ausfallpfade betrug in diesem ersten Experiment 5 %. Das heißt, 5 % aller Pfade (864 Pfade; alle 36 Pfade jeder Versuchsperson) waren fehlerhaft in dem Sinne, dass zum Beispiel ein falsches Symbol besucht oder notiert wurde, dass Symbole vergessen oder dass Symbole zuviel markiert wurden. Dabei produzierten männliche Versuchspersonen durchschnittlich 5,25 % Ausfallpfade, nur gering weniger als weibliche mit im Schnitt 5,37 %.

6 Ergebnisse Experiment 2

In diesem Kapitel werden die Ergebnisse dargestellt, die aus dem zweiten Experiment erhalten wurden. Dabei wurde, wie im ersten Experiment, zum einen ein Blick auf die gemessene Startzeit geworfen, die Versuchspersonen benötigten, zwischen Betrachten der *shopping* Liste und beginnen der Navigation. Der Hauptteil der Auswertung der Ergebnisse beruhte jedoch auch hier auf den produzierten Fehlern, dem *Overshoot*, und der Anzahl der Pfade, die mit der optimalen Lösung übereinstimmten.

Im zweiten Experiment soll auf eine verstärkte Verwendung einer regionen-basierten Strategie bei erschwerten Bedingungen geachtet werden, sowie auf die Rolle des Ortsgedächtnisses bei Pfadplanungsaufgaben. Ein ausgeprägtes Ortsgedächtnis sollte dazu führen, dass sich die Versuchspersonen die Umgebung schneller einprägen können und die Lösung der TSP-Aufgaben zu keinem starken Anstieg des *Overshoots* führt (Kapitel 3, Abschnitt 3.4).

Hierzu wurde das erste Experiment so modifiziert, dass jedes der 25 Symbole mit einer Abdeckung verdeckt werden kann. Die Versuchspersonen bekamen zu Beginn des Experiments Gelegenheit sich die Umgebung 10 min lang einzuprägen. Dann wurden die Symbole verdeckt und es wurde vor den eigentlichen TSP-Aufgaben ein Training absolviert. Wurde hierbei ein vorher festgelegtes Kriterium erfüllt, das heißt, wurden die letzten 5 Aufgaben des Trainings fehlerfrei gelöst, konnte sofort mit den TSP-Aufgaben begonnen. Wurde das Kriterium nicht erfüllt, bekamen die Versuchspersonen noch einmal die Gelegenheit sich das Feld 5 min lang einzuprägen. Der Ablauf der Navigationsaufgaben wurde wie im ersten Experiment durchgeführt. Nach jeder gelösten TSP-Aufgabe wurde das Feld erneut vollständig abgedeckt.

Erwartet wird ein Anstieg der Verwendung der regionen-basierten Strategie, da sich die Regionen schneller im Gedächtnis einprägen, als jeder einzelne Punkt auf dem Gitter und der kognitive Aufwand gering gehalten werden kann. Das Erlernen der Regionen lässt erwarten, dass die Symbole innerhalb der Regionen sich schneller im Gitter lokalisieren lassen, als bei Vernachlässigung der Regionen. Es wird ein geringerer *Overshoot* bei regionalen Pfaden als bei nichtregionalen Pfaden erwartet. Die Rolle des Ortsgedächtnisses sollte dazu führen, dass sich, je häufiger eine Versuchsperson die TSP-Aufgabe löst, eine Verbesserung der Lösungsqualität einstellt, da der Information zur Lage der Symbole viel mehr Aufmerksamkeit zuteil wird (Kapitel 3, Abschnitt 3.3).

6.1 Trainingsdaten

Es gab zwei Trainingsläufe, die die Versuchspersonen absolvieren mussten. Zuerst wurden ihnen 13 *shopping* Listen mit jeweils einem Startpunkt und einem Zielpunkt gezeigt. Diese Punkte sollten im Feld aufgedeckt werden. Die Anzahl der falsch aufgedeckten Symbole wurde notiert und das Feld nach jeder Aufgabe erneut vollständig abgedeckt. Danach wurden den Versuchspersonen 9 *shopping* Listen mit jeweils einem Startpunkt und zwei Zielpunkten gezeigt. Wieder sollten die Punkte im Feld aufgedeckt werden und wieder wurde die Anzahl der falsch aufgedeckten Symbole notiert. Das Feld wurde nach jeder Aufgabe vollständig abgedeckt. In Abbildung 6.1 ist die durchschnittliche Anzahl an Fehlern der Versuchspersonen für die einzelnen Trainingsläufe aufgezeigt.

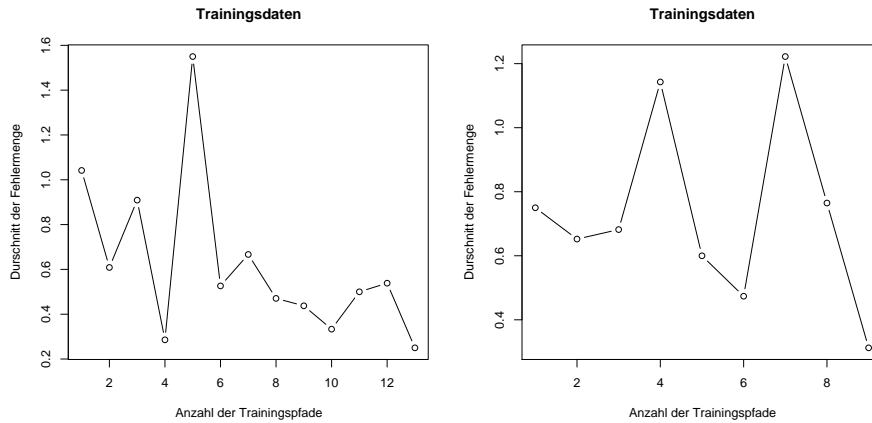


Abbildung 6.1: Durchschnitt der Fehler in den zwei Trainingsläufen.
 links: erster Trainingslauf mit 13 *shopping* Listen.
 rechts: zweiter Trainingslauf mit 9 *shopping* Listen.

Hatten die Versuchspersonen die letzten 5 Trainingsaufgaben fehlerfrei absolviert, konnte ihnen sofort die TSP-Aufgaben gestellt werden. Ansonsten durfte das Feld noch einmal einige Minuten betrachtet werden, bevor die Navigationsaufgaben gestellt wurden.

6.2 Auswertung 1

Das zweite Experiment wurde in zwei Auswertungen unterteilt. In die erste Auswertung gingen nur die Pfade ein, die die richtige Anzahl und nur korrekte Zielpunkte enthielten. Das heißt, Pfade bei denen die Zielpunkte nicht den Zielpunkten auf der *shopping* Liste entsprachen, und die mehr oder weniger Zielpunkte als vorgegeben enthielten, da diese von den Versuchspersonen falsch aufgedeckt oder vergessen wurden, gingen nicht in diese Auswertung mit ein. Wie in Kapitel 4 schon erwähnt, wurden nur die Pfade aus der Auswertung genommen, die Startzeiten nicht. Dies bedeutete, dass nicht nur, wie im ersten Experiment (Kapitel 5, Abschnitt 5.9), die Pfade wegfielen, bei denen nicht ersichtlich war, ob Versuchsperson oder Experimentator einen Fehler gemacht hatten, sondern auch diejenigen, bei denen falsche Zielpunkte zusätzlich zu den zu Besuchenden aufgedeckt wurden.

6.2.1 Startzeit

Die Startzeit ist die Zeit, die die Versuchsperson vom Betrachten der *shopping* Liste bis zum Beginn der Navigation benötigte. Die durchschnittliche Startzeit lag bei 13,07 Sekunden. Es konnte kein erheblicher Unterschied zwischen der regionalen Pfadgruppe mit durchschnittlich 12,94 Sekunden und der nichtregionalen Pfadgruppe mit durchschnittlich 13,21 Sekunden festgestellt werden (Abbildung 6.2.1 links). Eine ANOVA zeigte zwischen den Pfadtypen keinen signifikanten Effekt ($F(1,23) = 0,192$; $p = 0,666$; $\eta^2 = 0,008$; $MSE = 0,880$).

Die Startzeit in Abhängigkeit der Zielpunkte gab nur einen leichten Aufschluss über einen Anstieg der Startzeit mit steigender Anzahl an Zielpunkten (Pearson's product-moment correlation: $p < 2 \cdot 10^{-13}$, $r = 0,76$) (Abbildung 6.2.1 rechts). Eine ANOVA zeigte hier einen signifikanten Effekt ($F(5,115) = 9,780$; $p = 0,000$; $\eta^2 = 0,298$; $MSE = 1608,628$). Die Pfadtypen schienen im Gegensatz zu der Anzahl der Zielpunkte keinen Einfluss auf die Startzeit zu haben.

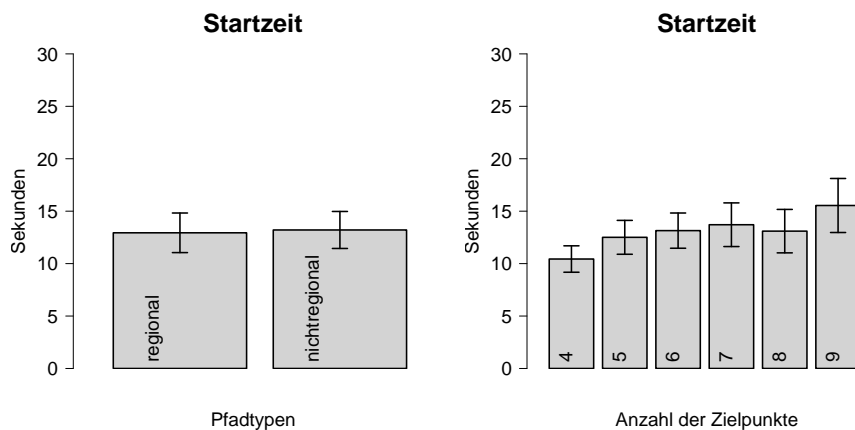


Abbildung 6.2.1: Startzeiten der Versuchspersonen.
 links: Startzeit bei regionalen und nichtregionalen Pfaden.
 rechts: Startzeiten nach Anzahl der zu besuchenden Zielpunkte.

Wie im ersten Experiment zeigte sich auch hier die Tendenz, dass weibliche Versuchspersonen eine schnellere Startzeit mit 11,11 Sekunden aufwiesen, als männliche mit 15,83 Sekunden. Ein deutlicher Unterschied zeigte sich zwischen den Geschlechtern bei regionalen Pfaden. Hier besaßen Männer durchschnittlich eine Startzeit von 16,07 Sekunden, Frauen eine von nur 10,70 Sekunden. Ein ähnliches Bild zeigt sich bei nichtregionalen Pfaden. Hier wurde eine Startzeit von 15,58 Sekunden bei Männern und eine von 11,51 Sekunden bei Frauen erhalten.

6.2.2 Fehlerquote (*Overshoot*)

Die Fehlerquote berechnete sich mittels der Formel 4.4 im vierten Kapitel. Die durchschnittliche Fehlerquote betrug 10,11 %. Die Fehlerquote bei nichtregionalen Pfaden lag mit 10,28 % nur knapp über der Fehlerquote der regionalen Pfade mit 9,94 % (Abbildung 6.2.2a). Eine ANOVA zeigte einen signifikanten Effekt bei der Anzahl an Zielpunkten ($F(5,115) = 11,013$; $p = 0,000$; $\eta^2 = 0,324$; $MSE = 2638,501$), während kein Effekt bei den Pfadtypen ($F(1,23) = 0,322$; $p = 0,576$; $\eta^2 = 0,014$; $MSE = 3,776$) gefunden wurde. Dies spiegelte sich auch im ersten Experiment wieder (Kapitel 5, Abschnitt 5.2). In Abhängigkeit der Zielpunkte konnte in diesem Experiment jedoch keine eindeutige Tendenz festgestellt werden. Es schien kein linearer Anstieg der Fehlerquote mit steigender Anzahl an Zielpunkten zu bestehen (Pearson's product-moment correlation: $p = 8,4 \cdot 10^{-13}$, $r = 0,23$). Die Anzahl der Zielpunkte schien einen Einfluss auf den *Overshoot* zu haben, während die Pfadtypen keinen Einfluss zeigten.

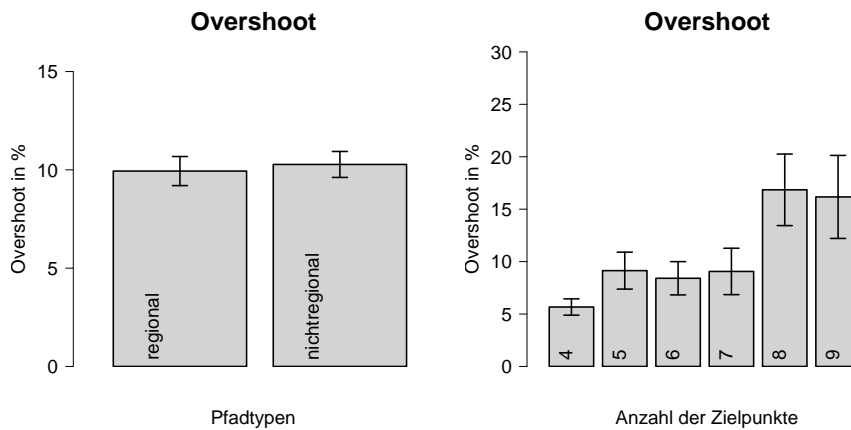


Abbildung 6.2.2a: Ermittelter *Overshoot* der ersten Auswertung. Einmal die beiden Pfadtypen gegeneinander (links) und einmal nach Anzahl der Zielpunkten (rechts).

Wie im ersten Experiment (Kapitel 5, Abschnitt 5.2) ließ sich auch hier eine Verbesserung der Fehlerquote im experimentellen Block 2 gegenüber dem experimentellen Block 1 erkennen (paired t-test: $t = 3,049$, $df = 23$, $p < 0,006$) (Abbildung 6.2.2b). Noch einmal zur Erklärung: Die 36 Pfade wurden in 2 experimentelle Hälften zu je 18 Pfaden eingeteilt. Die TSP-Aufgaben einer der beiden Hälften wurden in der ersten experimentellen Phase gelöst, die andere in der zweiten Phase. Wie im ersten Experiment hatten die experimentellen Blöcke einfluss auf den *Overshoot*.

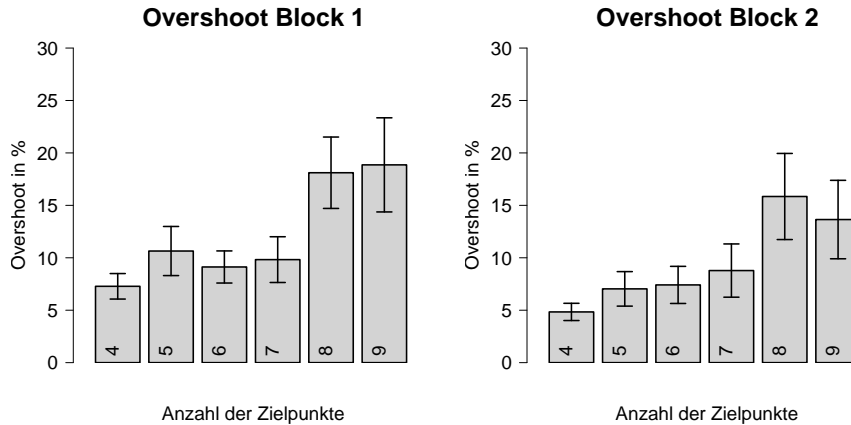


Abbildung 6.2.2b: *Overshoot* der beiden experimentellen Blöcke.

Des Weiteren zeigte sich kein großer Unterschied in der Fehlerquote zwischen regionalen und nichtregionalen Pfaden in Abhängigkeit der Zielpunkte. Es zeigte sich jedoch ein Anstieg des *Overshoots* bei 8 und 9 Zielpunkten (Abbildung 6.2.2c).

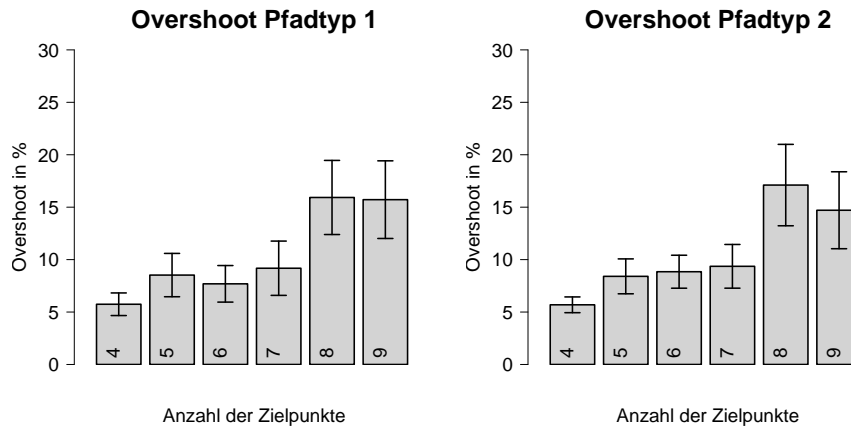


Abbildung 6.2.2c: *Overshoot* der beiden Pfadtypen.
 links: regionaler Pfadtyp.
 rechts: nichtregionaler Pfadtyp.

Zwischen weiblichen und männlichen Versuchspersonen zeigte sich ein etwas anderes Bild als in Experiment 1 (Kapitel 5, Abschnitt 5.2). In diesem Experiment produzierten männliche Versuchspersonen im Durchschnitt einen viel geringeren *Overshoot* als weibliche (Tabelle 6.2.2A).

Geschlecht	insgesamt	regional	nichtregional
männlich	Ø 8,86 %	Ø 8,00 %	Ø 9,73 %
weiblich	Ø 11,83 %	Ø 11,96 %	Ø 11,69 %

Tabelle 6.2.2A: *Overshoot* zwischen männlichen und weiblichen Versuchspersonen.

6.2.3 Korrekte Pfade

Die durchschnittliche Anzahl an korrekt gefundenen Pfaden lag bei 21,84 %. Das heißt, ungefähr 22 % der gelösten TSP-Aufgaben entsprachen der optimalen Lösung, die zuvor berechnet wurde. In der regionalen Pfadgruppe wurden durchschnittlich mit 29,92 % mehr korrekte Pfade gefunden als in der nichtregionalen Pfadgruppe mit 13,69 % (Abbildung 6.2.3 links). Eine ANOVA zeigte hier einen signifikanten Effekt ($F(1,23) = 6,756$; $p = 0,000$; $\eta^2 = 0,227$; $MSE = 0,546$). Somit schienen die Pfadtypen einen Einfluss auf die Anzahl der korrekt gefundenen Pfade aufzuweisen. Beim Blick auf die Anzahl der korrekt gefundenen Pfade in Abhängigkeit der Anzahl an Zielpunkten ließ sich keine eindeutige Aussage über eine Abhängigkeit der Pfadqualität in Hinsicht auf die Anzahl der Zielpunkte erkennen. Die Anzahl der korrekt gefundenen Pfade war bei etwa allen Pfaden gleich schlecht. Einzige Ausnahme schienen Pfade mit 6 Zielpunkten zu sein, deren durchschnittliche Anzahl an korrekt gefundenen Pfaden unter der der anderen Pfade lag (Abbildung 6.2.3 rechts). Eine ANOVA zeigte zwischen den Zielpunkten keinen signifikanten Effekt ($F(5,115) = 0,829$; $p = 0,532$; $\eta^2 = 0,035$; $MSE = 1,684$). Hier zeigte sich kein Einfluss der Anzahl der Zielpunkte auf die Anzahl der korrekt gefundenen Pfade.

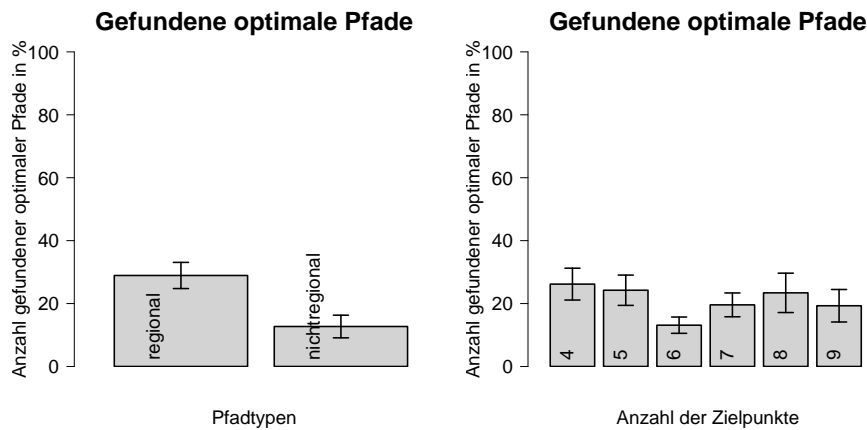


Abbildung 6.2.3: Anzahl der korrekt gefunden Pfade zwischen den beiden Pfadtypen (links) und nach Anzahl der Zielpunkte (rechts).

Es trat eine starke Signifikanz zwischen regionalen und nichtregionalen Pfaden auf (Welch Two Sample t-test: $t = 2,95$, $df = 45,075$, $p = 0,005$; paired t-test: $t = 5,48$, $df = 23$, $p = 1,432^{-05}$). Wie im ersten Experiment ließ sich hier ein Effekt zwischen regionalen und nichtregionalen Pfaden erkennen. Des Weiteren ließ sich keine Tendenz erkennen, die eine Abnahme der Anzahl an korrekt gefundenen Pfaden mit steigender Zahl an Zielpunkten aufwies (Pearson's product-moment correlation: $p = 0,01$, $r = 0,23$).

Wie auch schon im ersten Experiment (Kaptiel 5, Abschnitt 5.3) fanden weibliche Probanden mit 16,93 % durchschnittlich weniger korrekte Pfade als männliche mit 26,07 %. Auch ließ sich ein deutlicher Unterschied zwischen regionalen und nichtregionalen Pfaden erkennen. Frauen lösten mit 25,73 % deutlich weniger regionale TSP-Aufgaben optimal, als dies noch im ersten Experiment (Kaptiel 5, Abschnitt 5.3) der Fall war. Die männliche Gruppe hingegen lag mit 33,40 % nur etwas unter den 39 % aus dem ersten Experiment. Bei nichtregionalen Pfaden war dieser Unterschied genauso deutlich zu erkennen. Frauen lösten diese TSP-Aufgaben nur mit 8,64 % optimal, männliche Versuchspersonen hingegen mit 18,41 %.

6.2.4 Überschreitungen der Regionengrenzen

Auch im zweiten Experiment wurden im Laufe einer Navigation Regionengrenzen überschritten. Um eine Aufgabe zu lösen, mussten die Versuchspersonen zwischenzeitlich die Regionen wechseln. Somit kann man errechnen, wie häufig die Versuchsperson die Regionen wechseln sollte um eine optimale Lösung der Aufgabe zu erhalten. Im Falle von nichtregionalen Pfaden bedeutete dies, dass die Anwendung einer regionen-basierten Strategie zu weniger oder einer gleichen Anzahl an Regionenüberschreitungen führte. Einige der nichtregionalen Pfade hatten die Eigenschaft, dass die Verwendung einer regionen-basierten Strategie zu einer optimalen Anzahl an Regionenüberschreitungen führte, jedoch nicht zu einer optimalen Lösung.

Bei der Lösung von nichtregionalen Pfad-Aufgaben überschritten die Versuchspersonen durchschnittlich 0,60 mal weniger die Regionengrenzen, als dass sie dies bei den optimalen Pfad-Lösungen tun sollten (Abbildung 6.2.4, Anhang B, Abbildung B.2a und B.2b). Eine ANOVA zeigte einen signifikanter Effekt zwischen den Pfadtypen ($F(1,23) = 56,299$; $p = 0,000$; $\eta^2 = 0,710$; $MSE = 1,366$), sowie zwischen den Zielpunkten ($F(5,115) = 5,127$; $p = 0,003$; $\eta^2 = 0,182$; $MSE = 0,359$). Somit bestand ein Einfluss der Pfadtypen und Anzahl der Zielpunkte auf die Anzahl an Regionenüberschreitungen.

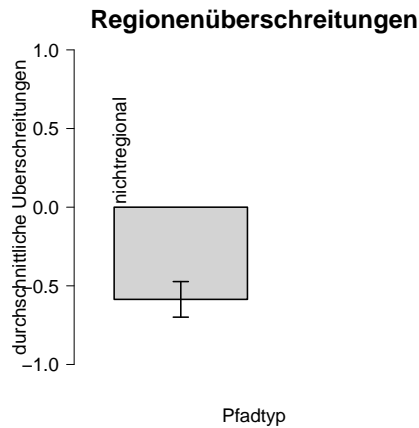


Abbildung 6.2.4: Durchschnittliche Anzahl der Regionenüberschreitungen bei nichtregionalen Pfaden.

Auch ein t-Test wies eine starke Signifikanz zwischen regionalen und nichtregionalen Pfaden auf (Welch Two Sample t-test: $t = 2,11$, $df = 44,616$, $p = 0,04$; paired t-test: $t = 4,93$, $df = 23$, $p = 5,5 \cdot 10^{-5}$; Pearson's product-moment correlation: $p = 2,51 \cdot 10^{-7}$, $r = 0,45$).

6.2.5 Regionen-basierte Strategie

Wie im ersten Experiment schon erwähnt (Kapitel 5, Abschnitt 5.5) konnte anhand der Regionenüberschreitungen berechnet werden, wie häufig eine regionen-basierte Strategie verwendet wurde. Dabei wurde zuerst für die optimale Lösung der 36 Pfade berechnet, wie häufig die Versuchsperson die Regionen wechseln musste, wenn sie eine regionen-basierte Strategie verwendete. Die Versuchsperson besuchte in diesem Fall jede Region nach der anderen. Dann wurde anhand der Pfade der Versuchspersonen die Anzahl an überquerten Regionengrenzen berechnet und mit der Anzahl an Regionenüberschreitungen der optimalen Lösung verglichen. Dies ergab, dass die Versuchspersonen die regionen-basierte Strategie in der regionalen Pfadgruppe durchschnittlich 0,48 mal, in der nichtregionalen Pfadgruppe durchschnittlich 0,14 mal verwendeten (Abbildung 6.2.5 links). Dieser Unterschied zeigte, dass auch in der nichtregionalen Pfadgruppe die regionen-basierte Strategie verwendet wurde, jedoch nicht so häufig, wie in der regionalen Pfadgruppe. Dabei folgten männliche Versuchspersonen der regionen-basierten Strategie bei beiden Pfadtypen häufiger, als weibliche.

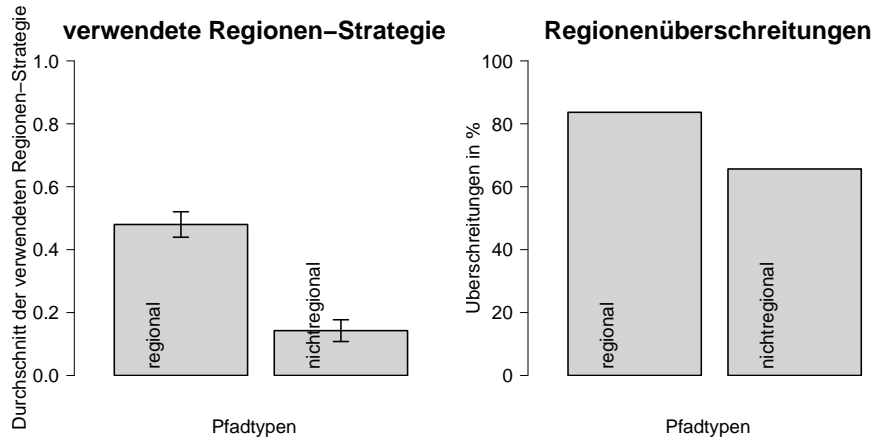


Abbildung 6.2.5: Regionen-basierte Strategie.
 links: Die im Durchschnitt verwendete Regionen-Strategie.
 rechts: Häufigkeiten der optimalen Regionenüberschreitungen.

Die Versuchspersonen lösten die TSP-Aufgaben mit der optimalen Anzahl an Regionenüberschreitungen in der regionalen Pfadgruppe mit 83,66 % sehr viel häufiger als in der Gruppe der nichtregionalen Pfade mit 64,64 % (Abbildung 6.2.5 rechts). Auch hier zeigte sich, dass die Versuchspersonen häufig die optimale Anzahl an Regionenüberschreitungen erreicht hatten.

6.2.6 Beste und schlechteste Pfade

Wie in Abschnitt 5.6 erwähnt, wurden die besten und schlechtesten Pfade anhand des *Overshoots* errechnet. Im zweiten Experiment erhielt man somit in der Gruppe der besten Pfade, den Pfad 14 aus der regionalen Gruppe und den Pfad 34 aus der nichtregionalen Gruppe. Unter die schlechtesten Pfade fielen die Pfade 17 und 20 aus der regionalen Gruppe und Pfad 15 aus der nichtregionalen Gruppe.

Durchschnittlich betrug die optimale Pfadlänge aller 36 Pfade eine Länge von 1397,36 cm. Die Pfadlänge der Versuchspersonen betrug im Schnitt 1536,32 cm, darunter die durchschnittliche Länge der regionalen Pfade mit 1527,49 cm und der nichtregionalen Pfade mit 1545,24 cm.

6.2.7 Gezeichnete Pfade der Versuchspersonen

Für jede Lösung der TSP-Aufgabe einer jeden Versuchsperson wurde der Pfad anhand der Reihenfolge der besuchten Zielpunkte rekonstruiert und alle dieser Lösungen übereinander in einer schematischen Abbildung der experimentellen Umgebung aufgezeichnet. In Abbildung 6.2.7 kann man erkennen, wie häufig die Versuchspersonen eine bestimmte Strecke zurückgelegt hatten. Dies zeigt sich an der Stärke der Linie. Je mehr Versuchspersonen eine Strecke zwischen zwei Punkten verwendeten, desto breiter zeigt sich die Linie in der Abbildung. In der linken Spalte sind die regionalen Pfade dargestellt und in der rechten Spalte die komplementären nichtregionalen Pfade.

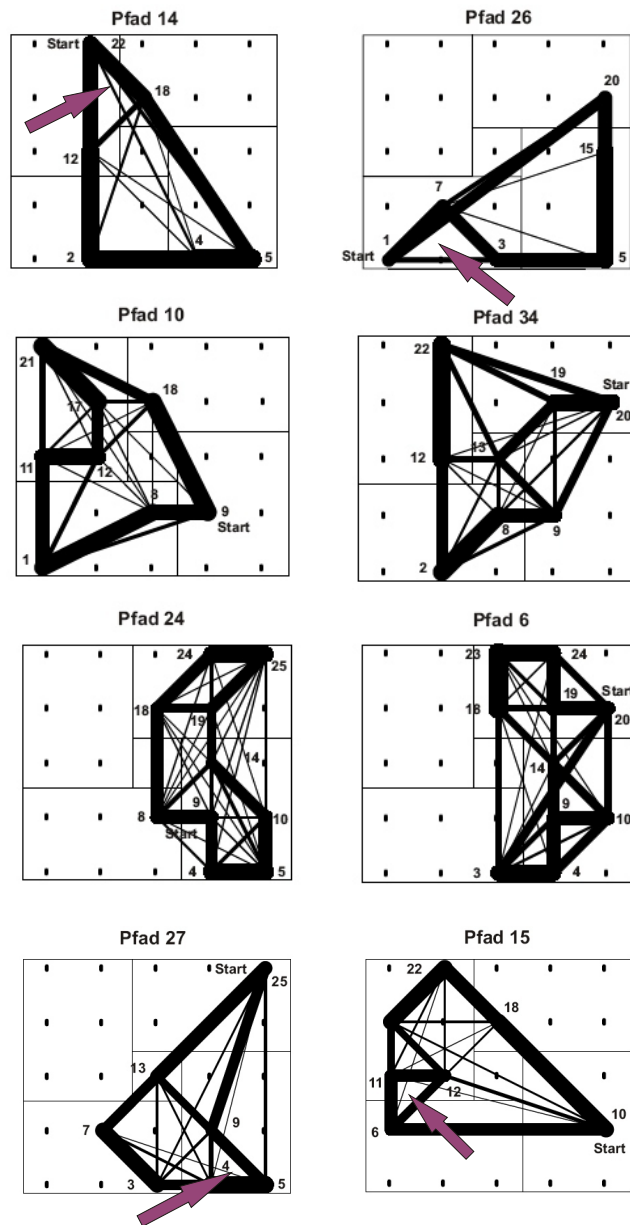


Abbildung 6.2.7: Pfade der Versuchspersonen aus der ersten Auswertung (links regionale Pfade, rechts nichtregionale Pfade).

In Abbildung 6.2.7 zeigt sich deutlich, ob eine regionen-basierte Strategie verwendet wurde und welche Auswirkungen dies auf den geplanten Pfad, speziell bei nichtregionalen Pfaden, hatte. In Pfad 14 ist deutlich zu erkennen, dass die meisten Versuchspersonen einer regionen-basierten Strategie folgten und deshalb auch häufig der optimale Pfad gefunden wurde. Im Gegensatz zu Pfad 26, indem eben diese Verwendung der regionen-basierten Strategie zu einem suboptimalen Pfad führte. Das gleiche Bild zeigt sich beispielsweise auch beim regionalen Pfad 27 und seinem komplementären nichtregionalen Pfad 15.

6.2.8 Kontrollfaktoren

Die Kontrollfaktoren hatten keinen erkennbaren Einfluss auf die Fehlerquote, die Startzeit oder das Finden der optimalen Lösung (Tabelle 6.2.8A und B).

Liste	Liste A	Liste B	t	df	p-Wert
<i>Overshoot</i> in %	Ø 9,74	Ø 11,45	0,43	22	0,67
Startzeit in Sekunden	Ø 13,72	Ø 12,43	0,35	22	0,73
korrekt gefundene Pfade in %	Ø 22,16	Ø 19,32	0,39	22	0,70

Tabelle 6.2.8A: Kontrollfaktor Feld. Zwei verschiedene Layouts des Feldes (Liste A und B).

Konfiguration	Konfiguration A	Konfiguration B	t	df	p-Wert
<i>Overshoot</i> in %	Ø 9,55	Ø 11,65	0,53	22	0,60
Startzeit in Sekunden	Ø 12,94	Ø 13,21	0,07	22	0,94
korrekt gefundene Pfade in %	Ø 21,49	Ø 19,98	0,21	22	0,84

Tabelle 6.2.8B: Kontrollfaktor Konfiguration. Zwei verschiedene Layouts der *shopping* Listen (Konfiguration A und B).

6.2.9 Ausfallpfade

Unter Ausfallpfaden versteht man die Pfade, in denen Fehler, wie falsche Zielpunkte, das Fehlen von Zielpunkten oder zu viel besuchte Zielpunkte auftraten. Diese Pfade wurden aus der Auswertung herausgenommen. Die Höhe der Ausfallpfade beträgt 15,74 %. Das heißt, 15,74 % aller Pfade waren fehlerhaft in der Weise, dass zum Beispiel ein falsches Symbol besucht oder notiert wurde, dass Symbole vergessen oder ein Symbole zuviel markiert wurden. Da es bei diesen Fehler teilweise nicht möglich war zurückzuverfolgen, ob Versuchsperson oder Experimentator den Fehler verursacht hatten, wurden diese Pfade aus der Auswertung herausgenommen. Männliche Versuchspersonen produzierten hier im Durchschnitt 14,72 % Ausfallpfade, und damit weniger als weibliche Versuchspersonen mit 17,26 %.

6.3 Auswertung 2

Wie in Abschnitt 6.2 erwähnt, wurde das zweite Experiment in zwei Auswertungen unterteilt. In die zweite Auswertung gingen nun auch die Pfade ein, die die richtige Anzahl und korrekte Zielpunkte enthalten, aber auch die Symbole, die während der Navigation falsch aufgedeckt wurden. Das heißt, Pfade bei denen Zielpunkte nicht den Zielpunkten auf der *shopping* Liste entsprachen, gingen nicht in diese Auswertung mit ein. Wie in Kapitel 4 schon erwähnt, wurden nur die Pfade aus der Auswertung genommen, die Startzeiten nicht. Dies bedeutete, dass im Gegensatz zur ersten Auswertung (Kapitel 6, Abschnitt 6.2), nur die Pfade wegfielen, bei denen nicht ersichtlich war, ob Versuchsperson oder Experimentator einen Fehler gemacht hatten. Diejenigen, bei denen falsche Zielpunkte zusätzlich zu den zu besuchenden aufgedeckt wurden, blieben in der Auswertung erhalten.

6.3.1 Startzeit

Die Startzeiten der zweiten Auswertung entsprachen den Startzeiten der ersten Auswertung in Abschnitt 6.2.1.

6.3.2 Fehlerquote (*Overshoot*)

Die Fehlerquote berechnete sich mittels der Formel 4.4 im vierten Kapitel. Die durchschnittliche Fehlerquote lag bei 12,81 %. Dabei lag die regionale Pfadgruppe mit einer Fehlerquote von durchschnittlich 12,61 % auf Höhe der nichtregionalen Pfadgruppe mit durchschnittlich 13 % (Abbildung 6.3.2a links). Wie zu erwarten war, lag diese Fehlerquote etwas über derjenigen in Auswertung 1 (Abschnitt 6.2.2). Eine ANOVA zeigte keinen signifikanten Effekt zwischen den Pfadtypen ($F(1,23) = 0,157$; $p = 0,695$; $\eta^2 = 0,007$; $MSE = 1,558$). Wie in der ersten Auswertung zeigten die Pfadtypen auch hier keinen Einfluss auf den *Overshoot*.

Die Betrachtung in Abhängigkeit der Anzahl der Zielpunkte ließ im Großen und Ganzen eine Erhöhung der Fehlerquote mit steigender Anzahl an Zielpunkten erkennen (Pearson's product-moment correlation: $p < 2 \cdot 10^{-16}$, $r = 0,67$). Auch eine ANOVA ließ hier einen signifikanten Effekt ($F(5,115) = 8,336$; $p = 0,000$; $\eta^2 = 0,266$; $MSE = 0,518$) erkennen (Abbildung 6.3.2a rechts). Somit hatte die Anzahl der Zielpunkte einfluss auf den *Overshoot*.

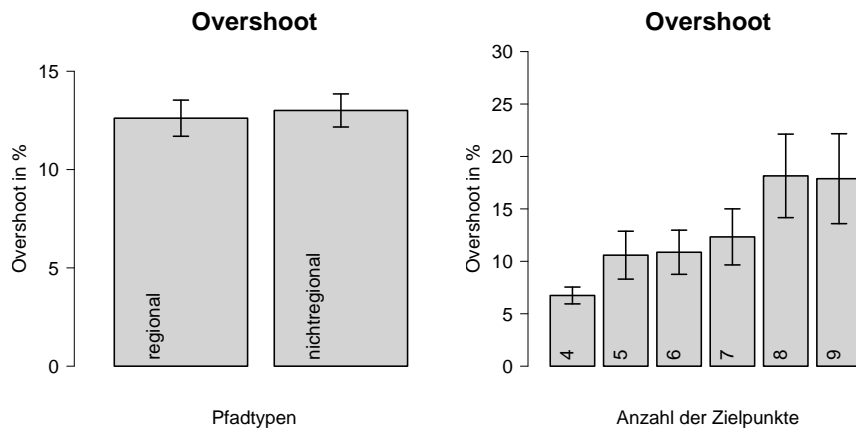


Abbildung 6.3.2a: Ermittelte *Overshoot* der zweiten Auswertung. Einmal die beiden Pfadtypen gegeneinander (links) und einmal nach Anzahl der Zielpunkten (rechts).

Wie auch schon in Abschnitt 6.2.2 zu sehen war, verbesserte sich die Fehlerquote im zweiten experimentellen Block ein wenig gegenüber dem ersten experimentellen Block (Abbildung 6.3.2b oben; paired t-test: $t = 3,79$, $df = 23$, $p < 0,001$). Auch hier zeigte sich der Einfluss der experimentellen Blöcke.

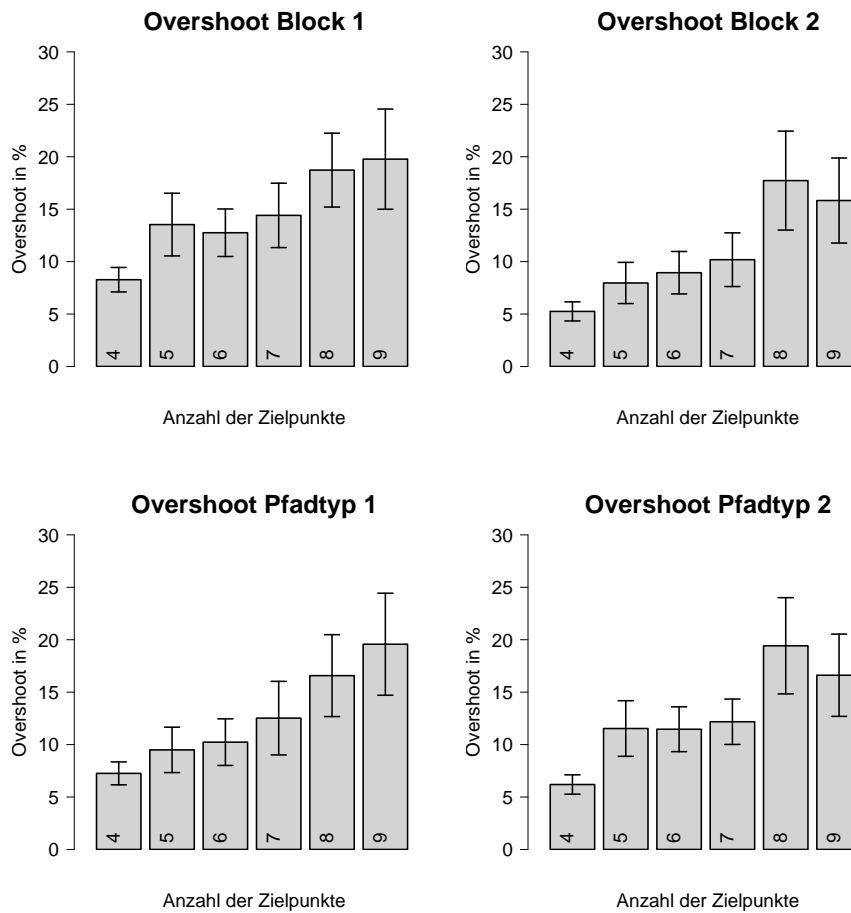


Abbildung 6.3.2b: *Overshoot*.

oben: *Overshoot* der experimentellen Blöcke.

unten: *Overshoot* der beiden Pfadtypen, regionale Pfade (links) und nichtregionale Pfade (rechts).

Beim Vergleich zwischen weiblichen und männlichen Versuchspersonen zeigte sich ein etwas anderes Bild als in Experiment 1 (Kapitel 5, Abschnitt 5.2). Männliche Versuchspersonen produzierten im Durchschnitt einen geringeren *Overshoot* als weibliche (Tabelle 6.3.2 A).

Geschlecht	insgesamt	regional	nichtregional
männlich	Ø 11,92 %	Ø 12,02 %	Ø 11,80 %
weiblich	Ø 13,72 %	Ø 13,86 %	Ø 13,59 %

Abbildung 6.3.2A: *Overshoot* zwischen männlichen und weiblichen Versuchspersonen.

6.3.3 Korrekte Pfade

Die Tendenz aus der ersten Auswertung (Abschnitt 6.2.3) zeigte sich auch hier. Die durchschnittliche Anzahl korrekt gefundener Pfade lag bei 19 % (Abbildung B.1 im Anhang). Das heißt, ungefähr 19 % der gelösten TSP-Aufgaben entsprachen der optimalen Lösung, die zuvor berechnet

wurde. Eine ANOVA zeigte einen signifikanten Effekt zwischen den Pfadtypen ($F(1,23) = 30,818$; $p = 0,000$; $\eta^2 = 0,573$; $MSE = 0,222$), sowie zwischen den Zielpunkten ($F(5,115) = 8,336$; $p = 0,000$; $\eta^2 = 0,266$; $MSE = 0,518$). Es schienen sowohl die Pfadtypen als auch die Anzahl der Zielpunkte einfluss auf die Anzahl der korrekt gefundenen Pfade zu haben.

6.3.4 Überschreitungen der Regionengrenzen

Im Laufe einer Navigation wurden Regionengrenzen überschritten, das heißt, um eine Aufgabe zu lösen, mussten die Versuchspersonen die Regionen wechseln. Wie in Abschnitt 6.2.4 erwähnt konnte man errechnen, wie oft die Versuchsperson die Regionen wechseln musste, wenn sie eine optimale Lösung der Aufgabe erhalten wollte. Auch hier zeigte sich dieser Effekt genauso deutlich, wie in der ersten Auswertung, mit durchschnittlich 0,57 mal weniger überschrittenen Regionengrenzen (Pearson's product-moment correlation: $p = 5,67^{-13}$, $r = 0,554$) (Abbildung B.2a und B.2b im Anhang B). Eine ANOVA zeigte einen signifikanten Effekt zwischen den Pfadtypen ($F(1,23) = 60,308$; $p = 0,000$; $\eta^2 = 0,724$; $MSE = 0,996$). Wie in der ersten Auswertung schienen die Pfadtypen einfluss auf die Anzahl der Regionenüberschreitungen zu haben.

6.3.5 Regionen-basierte Strategie

Wie im ersten Experiment schon erwähnt (Kapitel 5, Abschnitt 5.5) konnte anhand der Regionenüberschreitungen berechnet werden, wie häufig eine regionen-basierte Strategie verwendet wurde. Die Versuchspersonen verwendeten in dieser Auswertung die regionen-basierte Strategie bei regionalen Pfaden durchschnittlich 0,41 mal und bei nichtregionalen Pfaden 0,13 mal (Abbildung 6.3.5 links).

Mit 80,33 % bei regionalen Pfaden folgten die Versuchspersonen häufiger den Regionenüberschreitungen der optimalen Pfadlösung als bei nichtregionalen Pfaden mit 64,52 % (Abbildung 6.3.5 rechts).

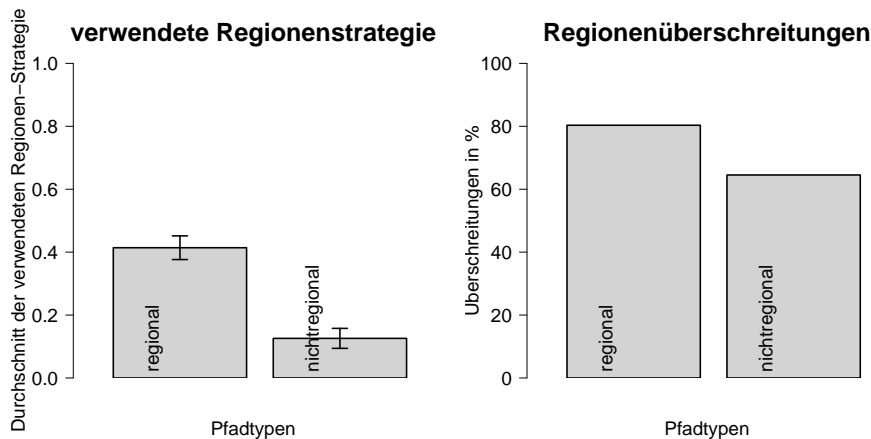


Abbildung 6.3.5: Regionen-basierte Strategie.
links: Die im Durchschnitt verwendete Regionen-Strategie.
rechts: Häufigkeiten der optimalen Regionenüberschreitungen.

Dabei folgten männliche Versuchspersonen der regionen-basierten Strategie bei beiden Pfadtypen häufiger, als weibliche (siehe Anhang B, Tabelle B.3A).

6.3.6 Beste und schlechteste Pfade

Die besten und schlechtesten Pfade wurden anhand des *Overshoots* berechnet (Kapitel 5, Abschnitt 5.6). Somit erhielt man in der Gruppe der besten Pfade, den Pfad 14 aus der regionalen Gruppe und den Pfad 34 aus der nichtregionalen Gruppe. Unter die schlechtesten Pfade fielen die Pfade 17 aus der regionalen Gruppe und Pfad 15 aus der nichtregionalen Gruppe.

Die Pfadlänge aller 36 optimal gelösten Pfade betrug im Schnitt 1397,36 cm. Die durchschnittliche Länge der Pfade betrug in dieser Auswertung 1578,88 cm. Dabei entfielen durchschnittlich 1575,17 cm auf die Länge der regionalen Pfade und 1582,61 cm auf die Länge der nichtregionalen Pfade.

6.3.7 Kontrollfaktoren

Die Kontrollfaktoren hatten keinen erkennbaren Einfluss auf die Fehlerquote, die Startzeit oder das Finden der optimalen Lösung (Tabelle 6.3.7A und B).

Liste	Liste A	Liste B	t	df	p-Wert
<i>Overshoot</i> in %	Ø 12,51	Ø 13,06	0,21	22	0,84
Startzeit in Sekunden	Ø 13,72	Ø 12,43	0,35	22	0,73
korrekt gefundene Pfade in %	Ø 20,47	Ø 17,47	0,44	22	0,66

Tabelle 6.3.7A: Kontrollfaktor Feld. Zwei verschiedene Layouts des Feldes (Liste A und B).

Konfiguration	Konfiguration A	Konfiguration B	t	df	p-Wert
<i>Overshoot</i> in %	Ø 12,51	Ø 13,06	0,11	22	0,92
Startzeit in Sekunden	Ø 12,94	Ø 13,21	0,07	22	0,94
korrekt gefundene Pfade in %	Ø 19,07	Ø 18,87	0,03	22	0,98

Tabelle 6.3.7B: Kontrollfaktor Konfiguration. Zwei verschiedene Layouts der *shopping* Listen (Konfiguration A und B).

6.3.8 Ausfallpfade

In dieser zweiten Auswertung versteht man unter Ausfallpfaden diejenigen Pfade, in denen das Fehlen von auf der *shopping* Liste enthaltenen Zielpunkten, auftraten. Da es bei diesen Fehler teilweise nicht möglich war zurückzuverfolgen, ob Versuchsperson oder Experimentator den Fehler verursacht hatte, wurden diese Pfade aus der Auswertung herausgenommen. Im Gegensatz zur ersten Auswertung (Abschnitt 6.9) wurden Pfade mit richtig besuchten Zielpunkten plus falsch aufgedeckten Zielpunkten nicht aus der Auswertung herausgenommen. Die Höhe der Ausfallpfade lag bei 3,36 %. Weibliche Versuchspersonen hatten mit 3,37 % durchschnittlich wiederum mehr Ausfallpfade, als männliche mit 1,39 %.

7 Diskussion

Allgemein wurde erwartet, dass die Lösungsqualität bei regionalen Pfaden deutlich höher liegt als bei nichtregionalen Pfaden. Die Verwendung einer regionen-basierten Strategie sollte zu optimalen Lösungen in der regionalen Pfadgruppe führen und dabei der kognitive Aufwand so gering wie möglich gehalten werden. Auch wurde erwartet, dass die Lösungsqualität mit steigender Zahl von Zielpunkten abnimmt. Erstens, da mehr Möglichkeiten existieren und diese bei der Pfadplanung beachtet werden müssen und zweitens, da mit steigender Zahl an Zielpunkten auch die Belastung des Arbeitsgedächtnisses ansteigt. An irgendeinem Punkt ist es nicht mehr möglich, gleichzeitig die Positionen aller Zielpunkte im Gedächtnis zu behalten, sodass diese Pfade nicht unter Betrachtung aller Zielpunkte geplant werden können (Abbildung 7).

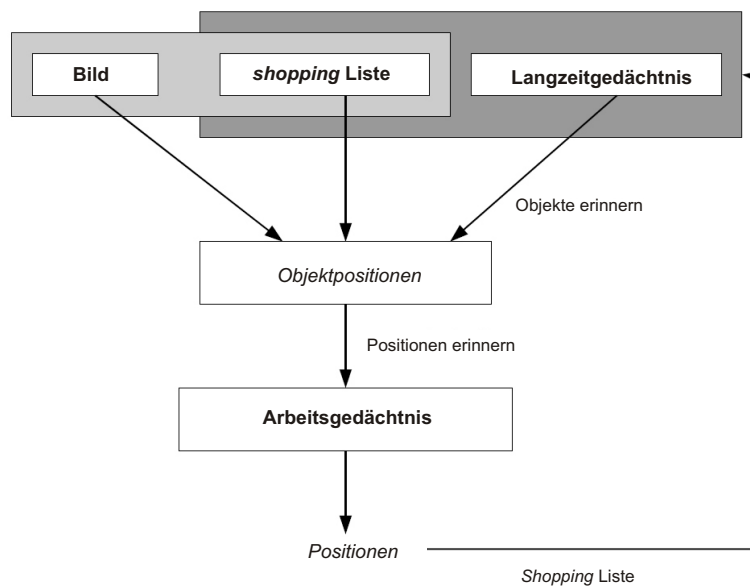


Abbildung 7: Ein Überblick über die Beziehungen zwischen dem Arbeitsgedächtnis, Ortsgedächtnis und den beiden zuvor vorgestellten Experimenten.

Deshalb wurde eine mögliche Strategie, die regionen-basierte Strategie, in den beiden Experimenten dieser Arbeit getestet. Bei manchen optimalen Pfaden wurden mehr Regionenüberschreitungen benötigt, als bei den dazugehörigen suboptimalen Pfaden. Diese suboptimalen Pfade wurden von den Versuchspersonen bevorzugt. Dieses Ergebnis deutet darauf hin, dass zuerst ein grober Pfad auf Regionenebene geplant und dieser dann während der Navigation verfeinert wurde. Solch eine hierarchische Planungsstrategie erlaubt es, den Rechenaufwand während der Planung zu reduzieren.

7.1 Experiment 1

Die Fehlerquote und Startzeit stieg mit steigender Anzahl an Zielpunkten. Der Unterschied in der Startzeit von männlichen und weiblichen Versuchspersonen lag wahrscheinlich daran, dass die Männer den Pfad so gut wie möglich im voraus planen und Frauen hier etwas intuitiver vorgehen (Kapitel 5, Abschnitt 5.1).

Im Gegensatz zu den vorher getroffenen Aussagen unterschied sich die Fehlerquote nicht deutlich zwischen der regionalen Pfadgruppe und der nichtregionalen Pfadgruppe. Regionale Pfade zeigten

dennoch eine etwas höhere Qualität als nichtregionale Pfade. Jedoch waren die Lösungen der nichtregionalen Pfade nur wenig schlechter als die der regionalen Pfade. Ein Anstieg des *Overshoots* in Abhängigkeit von einer steigenden Anzahl an Zielpunkten scheint darauf hinzudeuten, dass ab einer bestimmten Anzahl an Informationseinheiten das Arbeitsgedächtnis nicht mehr in der Lage ist, jeder Information die gleiche Aufmerksamkeit zukommen zu lassen (Kapitel 3, Abschnitt 3.1). Deshalb fällt es bei einer steigenden Anzahl an Zielpunkten zunehmend schwerer, alle Punkte im Gedächtnis zu behalten um damit einen optimalen Pfad zu planen. Die Rolle des Ortsgedächtnisses zeigt sich deutlich im Vergleich der beiden experimentellen Blöcke. Ein Absinken des *Overshoots* im zweiten experimentellen Block deutet darauf hin, dass die Aufmerksamkeit, die der Information zur Lage der Objekte gewidmet wurde, zu einer Verbesserung der Pfadqualität führt. Die Auswertung des Navigationsverhaltens der Versuchspersonen auf Regionenebene legte aussagekräftige Unterschiede zwischen den Pfadtypen offen. Auf Regionenebene machten die Versuchspersonen weniger Fehler bei regionalen Pfaden, als bei nichtregionalen Pfaden (Abschnitt 5.2). In Kombination mit der Tatsache, dass die Versuchspersonen weniger Regionengrenzen bei nichtregionalen Pfaden überschritten, als sie im optimalen Pfad sollten, deutet stark auf die Verwendung einer regionen-basierten Strategie hin. Zu beachten ist, dass bei nichtregionalen Pfaden die Anwendung einer regionen-basierten Strategie nicht nur zu suboptimalen Pfaden führt, sondern auch zu Pfaden, bei denen weniger oft die Regionen gewechselt wurden, als das im optimalen Pfad der Fall sein sollte. Die Versuchspersonen nahmen hier Umwege in Kauf, indem sie eine regionen-basierte Strategie verwendeten, auch wenn dies zu suboptimalen Pfaden führte (Abschnitt 5.4). Aus Abbildung 5.5 im Abschnitt 5.5 ist ersichtlich, dass die Versuchspersonen häufig die optimale Anzahl an Regionenüberschreitungen erreicht hatten, wie dies anhand des optimalen Pfades berechnet wurde. Dies ließ jedoch nicht den Schluss zu, dass damit auch die optimale Lösung gefunden wurde, da Fehler sehr häufig nicht bei der Verwendung einer regionen-basierten Strategie auftraten, sondern innerhalb der Regionen. Die hohe Anzahl von 62,41 % an optimalen Regionenüberschreitungen in der nichtregionalen Pfadgruppe deutete darauf hin, dass nicht immer eine regionen-basierte Strategie verwendet wurde, da hier ein sehr viel geringerer Effekt erwartet wurde, falls die Versuchspersonen konsequent diese Strategie verfolgt hätten.

Die fehlende Differenz in der Fehlerquote zwischen den Pfadtypen entstand, da auch Fehler innerhalb der Regionenebene auftraten und somit zu suboptimalen Pfaden führten, obwohl die Versuchspersonen eine regionen-basierte Strategie zur Lösung des TSPs verwendeten. Dies führt zu der Annahme, dass die Versuchspersonen zuerst einen groben Pfad auf Regionenebene planten und diesen während der Navigation verfeinerten. Während der Navigation konnten dabei Fehler innerhalb der Regionen auftreten. Es besteht auch die Möglichkeit, dass die Regionen in einer suboptimalen Reihenfolge besucht wurden und somit den *Overshoot* noch verstärkten. Weitere Möglichkeiten in Hinsicht des *Overshoots* sind die geringen Distanzen im Gitter. Bei größeren Fehlern resultiert dadurch kein größerer Anstieg des *Overshoots*. Desweiteren können Zielpunkte während der Navigation vergessen worden sein und mussten dann in die schon zurückgelegte Teilstrecke miteingebaut werden. Das eine, wie das andere, konnte in beiden Pfadgruppen zu einem *Overshoot* führen. Die Verbesserung der Fehlerquote von der experimentellen Phase 1 zur experimentellen Phase 2 (Abbildung 5.2a, Abschnitt 5.2) kann damit erklärt werden, dass es einfacher wird die Pfade annähernd optimal zu planen, je länger sich die Versuchsperson innerhalb der experimentellen Umgebung bewegte und je mehr TSP-Aufgaben gelöst wurden. Somit lässt sich hier der Einfluss des Ortsgedächtnisses deutlich erkennen (Kapitel 3, Abschnitt 3.3)

Zusammenfassend lässt sich sagen, dass die Versuchspersonen in den meisten Fällen eine regionen-basierte Strategie verwendeten, jedoch Fehler innerhalb der Regionen machten. Führt dies zu einer zu hohen Fehlerquote, sollte diese Strategie verworfen und nicht mehr verwendet werden.

7.2 Experiment 2

Auch hier stiegen Fehlerquote und Startzeit mit ansteigender Zahl an Zielpunkten. Die etwas geringere Startzeit im zweiten Experiment (Kapitel 6, Abschnitt 6.2.1) scheint daran zu liegen, dass die Versuchspersonen schneller einen groben Pfad über die Punkte planten, die sie sicher

aus dem Langzeitgedächtnis abrufen konnten. Punkte, die die Versuchsperson nicht so sicher im Gedächtnis behalten hatte, wurden während der Navigation mit in den Pfad eingebaut, auch wenn damit ein größerer Zeitverlust während der Navigation verbunden war.

Wie schon im ersten Experiment (Kapitel 5, Abschnitt 5.2) unterschied sich die Fehlerquote nicht zwischen regionalen und nichtregionalen Pfaden. Der erhöhte *Overshoot* und die Höhe der Ausfallpfade zeigte einen Abfall der Qualität der Lösungen. Dies ist auch an einer niedrigeren Anzahl an korrekt gefundenen Pfaden zu erkennen. Auch hier zeigte sich eine Verbesserung von der ersten experimentellen Phase zur zweiten experimentellen Phase, was wiederum auf die Rolle des Ortsgedächtnisses hindeutet. Die Stärke der Aufmerksamkeit für eine Information scheint auch hier zu einem Absinken des *Overshoots* zu führen. Wiederum lässt sich dies damit erklären, dass die Versuchspersonen einen annähernd optimalen Pfad planten, je länger sie sich in der experimentellen Umgebung bewegten, und damit auch in Hinsicht der Lokalisation von Zielpunkten sicherer wurden und vor allem, desto mehr TSP-Aufgaben gelöst wurden. Auch hier zeigten die Lösungen der regionalen Pfaden eine bessere Qualität als die Lösungen der nichtregionalen Pfade.

Wie im vorigen Abschnitt 7.1 schon erwähnt, überschritten die Versuchspersonen auch im zweiten Experiment weit weniger Regionengrenzen bei nichtregionalen Pfaden, was wiederum zu einer suboptimalen Lösung führte und die Annahme der Verwendung einer regionen-basierten Strategie aus dem ersten Experiment unterstützt. Die Verwendung einer regionen-basierten Strategie führte jedoch nicht immer zu optimalen Pfaden, da die meisten Fehler innerhalb der Regionen auftraten. Die Fehlersequenz innerhalb der Pfade scheint anzusteigen, je unsicherer sich die Versuchspersonen bei der Lage der Zielpunkte sind. Hierbei wird der zuerst gebildete grobe Pfad nicht nur über die Regionen aufgebaut, sondern auch über die Sicherheit der Versuchsperson in Bezug auf die Lage der Zielpunkte. Dies kann den Effekt in Hinsicht auf einen suboptimalen Pfad noch verstärken und führte, wie im zweiten Experiment zu sehen war, zu einer höheren Fehlerquote im Vergleich zum ersten Experiment. Dies kann auch zu einer weniger häufigen Verwendung der regionen-basierten Strategie führen, als erwartet.

Der Vergleich der beiden Auswertungen in Experiment 2 (Kapitel 6, Abschnitte 6.2 und 6.3) führte zu keinem größeren Unterschied. Der etwas höhere *Overshoot* der zweiten Auswertung (Abschnitt 6.3.2) war zu erwarten, da in diese Auswertung alle Pfade mit zusätzlich falsch aufgedeckten Zielpunkten mit eingingen. Eine ANOVA zeigte zwischen den beiden Auswertungen einen signifikanten Effekt ($F(1,23) = 7,910$; $p = 0,010$; $\eta^2 = 0,256$; $MSE = 115,682$). Jedoch ist der Unterschied von 3 % geringer als erwartet und scheint daran zu liegen, dass die Versuchspersonen sehr viel weniger Fehler gemacht hatten, als im Vorfeld erwartet wurde. Der geringe Anstieg des *Overshoots* kann, wie in Abschnitt 7.1 schon erwähnt, an den geringen Distanzen im Gitter liegen, die bei Fehlern nicht zu einem deutlichen Anstieg der Fehlerquote führten. Der große Unterschied von 12 % bei den Ausfallpfaden (Abschnitt 6.2.9 und 6.3.8) der beiden Auswertungen zeigte hier jedoch deutlich, dass viele Pfade mit falsch aufgedeckten Zielpunkten von den Versuchspersonen produziert wurden. Auch die etwas geringere Anzahl an gefundenen optimalen Pfaden in der zweiten Auswertung und die einigermaßen gleich häufige Verwendung der regionen-basierten Strategie, zeigte hier die vorhergesagte Tendenz. Eine ANOVA konnte hier bei den korrekt gefundenen Pfaden einen signifikanten Effekt zwischen den beiden Experimenten finden ($F(1,23) = 12,175$; $p = 0,002$; $\eta^2 = 0,346$; $MSE = 0,004$), sowie bei der Verwendung einer regionen-basierten Strategie ($F(1,23) = 42,712$; $p = 0,000$; $\eta^2 = 0,650$; $MSE = 0,019$).

Die Aufgabe wurde in diesem Experiment hinsichtlich des ersten Experiments erschwert, indem die Versuchspersonen das Feld erst lernen mussten um die Aufgabe zu lösen. Dabei spielt die Gedächtnisleistung eine wichtige Rolle, um die Fehlerquote so gering wie möglich zu halten (siehe Kapitel 3, Abschnitt 3.4). Da jede Versuchsperson eine andere Art des Lernens aufweist, kann hier keine eindeutige Aussage über eine bestimmte Lernstrategie gemacht werden. Es zeigte sich jedoch, dass Versuchspersonen, deren Gedächtnis in Bezug auf "memory"-Aufgaben ausgeprägt zu sein scheinen, einen schnelleren und effektiveren Lernweg aufzeigten und sich in der experimentellen Umgebung besser und schneller zurecht fanden als andere.

Auch im zweiten Experiment lässt sich sagen, dass die Versuchspersonen in den meisten Fällen eine

regionen-basierte Strategie verwendeten, jedoch Fehler innerhalb der Regionen machten. Führt dies zu einer hohen Fehlerquote, so sollte die Strategie verworfen und nicht weiter verwendet werden. Hier fließt jedoch noch die Fähigkeit des Lernens jeder Versuchsperson mit ein. Versuchspersonen mit einer besseren Gedächtnisleistung oder einer optimaleren Lernstrategie als andere, produzierten einen geringeren *Overshoot*. Deshalb entscheidet zusätzlich die Art des Lernens, ob eine Strategie verwendet werden sollte, oder nicht.

7.3 Vergleich der beiden Experimente

Zwischen dem ersten und zweiten Experiment zeigte sich eine Abnahme der Startzeit. Wie in Abschnitt 7.2 erwähnt, konnte dies an einer schnelleren Planung des Pfades liegen, die einfacher wurde, sobald die Versuchspersonen nicht alle Punkte des Feldes visuell vor sich hatten, sondern sie aus dem Langzeitgedächtnis abrufen mussten und somit den Pfad zuerst grob über die Punkte planten, bei denen sie sicher deren Standpunkt lokalisieren konnten. Eine ANOVA zeigte hier jedoch keinen signifikanten Effekt zwischen den beiden Experimenten ($F(1,23) = 0,992$; $p = 0,330$; $\eta^2 = 0,041$; $MSE = 114,941$).

Wie erwartet zeigte sich im zweiten Experiment ein Anstieg der Fehlerquote. Eine ANOVA zeigte zwischen den beiden Experimenten einen signifikanten Effekt ($F(1,23) = 7,149$; $p = 0,014$; $\eta^2 = 0,237$; $MSE = 743,634$). Ein Ansteigen der Fehlerquote scheint daran zu liegen, dass es schwieriger wird eine TSP-Aufgabe zu planen, sobald außer dem Arbeitsgedächtnis eine Interaktion mit dem Langzeitgedächtnis benötigt wird. Dabei kommt es auch darauf an, wie schnell und effektiv die Versuchspersonen sich die Objekte im Langzeitgedächtnis eingeprägt haben. Jedoch ist der Unterschied zwischen den Fehlerquoten geringer als erwartet und lässt vermuten, dass kein starker Anstieg des *Overshoots* zu erwarten ist, wenn über die geringe Zunahme der Pfadlänge durch geringe Distanzen im Gitter, ein suboptimaler Pfad zu einem kleinen Umweg führte.

Die Anzahl der optimal gefundenen Pfade sank im zweiten Experiment etwa um die Hälfte (Abschnitt 5.3, 6.2.3 und 6.3.3). Es scheint schwieriger zu werden, den optimalen Pfad zu bilden, wenn alle zu besuchenden Zielpunkte nicht nur im Arbeitsgedächtnis behalten werden müssen, sondern auch noch zusätzlich eine Projektion der Lokalisation der Punkte im Gitter erstellt werden muss, um die Aufgabe zu lösen. Eine ANOVA zeigte in Bezug auf die korrekt gefundenen Pfade einen signifikanten Effekt zwischen den beiden Experimenten ($F(1,23) = 4,673$; $p = 0,041$; $\eta^2 = 0,169$; $MSE = 0,101$). Um den optimalen Pfad zu finden, scheint jedoch die Aufteilung des Gitters in Regionen ein Hilfe zu sein, indem eine regionen-basierte Strategie verwendet wird. Dies wird aus den Abschnitten 5.4 und 5.5, sowie 6.2.4, 6.2.5, 6.3.4 und 6.3.5 ersichtlich. Hier zeigte eine ANOVA keinen signifikanten Effekt zwischen den beiden Experimenten ($F(1,23) = 3,901$; $p = 0,060$; $\eta^2 = 0,237$; $MSE = 0,073$).

Zu erkennen ist auch aus den Abschnitten 5.6, 6.2.6 und 6.3.6, dass es Unterschiede gibt, wie schwer eine TSP-Aufgabe zu lösen ist. Dabei schien der Pfad 15 eine große Rolle zu spielen, da er in beiden Experimenten der Pfad mit dem höchsten *Overshoot* war. Des Weiteren zeigt die Höhe der Ausfallpfade in Abschnitt 5.9 und 6.2.9, dass es für die Versuchspersonen nicht so leicht war Pfade zu planen, für die die Verwendung des Arbeitsgedächtnisses und des Ortsgedächtnisses nötig war.

Wie in Abschnitt 7.2 schon erwähnt, spielt dabei die Art des Lernens eine wichtige Rolle, die darüber entscheidet, wie schnell eine Information in das Langzeitgedächtnis überführt wird, um eine Projektion dieser Information schnell und immer dann wenn sie benötigt wird wieder abzurufen.

8 Vergleich zwischen existierenden Heuristiken und menschlicher Wegeplanung

”A heuristic is the analysis of the instruments and ways of solving tasks” (G. Pólya).⁹ Eine Heuristik ist eine intuitive Methode, deren vorgegebene Eigenschaften nicht erprobt sind. In der Mathematik werden solche Methoden verwendet, falls keine anderen Methoden für ein Problem bekannt sind, welche diese in polynomieller Zeit lösen können.

Heuristiken sind nicht immer erfolgsversprechend. Im allgemeinen geben sie eine annähernd gute Lösung, können jedoch keine exakte Lösung garantieren. Meistens ist die Qualität der Lösung von den verwendeten Parametern abhängig. Heuristiken sind dazu bestimmt, modifiziert zu werden, das bedeutet, sie sind keine fertigen Algorithmen, können aber zu solchen werden. Die Suche nach einer einzelnen Heuristik führt meist dazu, dass eine Menge an ähnlichen Heuristiken gefunden wird. Diese unterscheiden sich nur in wenigen Eigenschaften voneinander [TANLER, 2004].

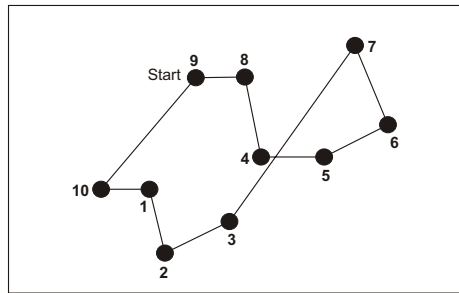
Viele annähernd gute heuristische Algorithmen wurden im Zusammenhang mit dem Traveling Salesperson Problem entwickelt und einige von ihnen könnten als Modelle für menschliche Verhaltensweisen zu dienen. In diesem Kapitel werden einige bekannte Algorithmen, die zur Lösung des TSP verwendet werden, vorgestellt. Es wird gezeigt, welche Algorithmen als Modell für das Verhalten der Versuchspersonen in Frage kommen könnten, indem das Verhalten der Versuchspersonen in den Experimenten mit der Vorgehensweise des Algorithmus zur Lösung des TSPs im Vergleich herangezogen wird.

8.1 Nearest Neighbour Algorithmus

Die ursprünglichste Heuristik für das Traveling Salesperson Problem ist der bekannte Nearest Neighbour (NN) Algorithmus. In diesem Algorithmus hat der Reisende die Vorgabe zum jeweils nächstgelegenen Punkt in Abhängigkeit zu seinem aktuellen Bezugspunkt zu reisen.

So wird für den Nearest Neighbour Algorithmus ein Startplatz, der sogenannte *starting point*, als Anfang des Pfades gewählt. Der am nächsten gelegene, noch nicht besuchte, Punkt vom Startpunkt aus, wird zum Pfad hinzugefügt und zum neuen *starting point* gewählt. Dies wird solange wiederholt, bis alle Knoten zum Pfad hinzugefügt wurden. Dann wird der letzte und erste Punkt zusammengefügt, um den Pfad zu vervollständigen. Es wird eine Anordnung $c_{\pi(1)}, \dots, c_{\pi(N)}$ der Punkte aufgebaut, in der der Startplatz $c_{\pi(1)}$ willkürlich gewählt wird und es wird der Punkt $c_{\pi(i+1)}$ als Punkt c_k gewählt, welcher die Distanz zwischen zwei Punkten $\{d(c_{\pi(i)}, c_k) : k \neq \pi(j), 1 \leq j \leq i\}$ minimiert. Der daraus entstehende Pfad verschiebt die Punkte in die vom Algorithmus neu gebildete Ordnung und kehrt zu Punkt $c_{\pi(1)}$ zurück, nach dem der letzte Punkt $c_{\pi(N)}$ besucht wurde (Abbildung 8.1). Der Nearest Neighbour berechnet $\frac{n \cdot (n-1)}{2}$ mal die Distanz zwischen zwei Punkten und benötigt eine Laufzeit von $\mathcal{O}(n^2)$ [JOHNSON, 1995].

⁹ ”Eine Heuristik ist die Untersuchung oder Auswertung der Instrumente und Methoden um Aufgaben zu lösen.”



insertion order: 9(8, 4, 5, 6, 7, 3, 2, 1, 10)

Abbildung 8.1: Beispiel eines Pfades, der mit dem Nearest Neighbour Algorithmus berechnet wurde.

Der Nearest Neighbour Algorithmus (Algorithmus 8.1) findet nicht immer die optimale Lösung, denn er kann verschiedene Pfade liefern, die abhängig vom gewählten Startpunkt sind.

```

function nearestNeighbour
begin
  choose startingNode  $v_{i_1}$ 
   $z := 0, j := 1;$ 

  for  $j = 2$  to  $n$  do
    if  $[v_{i_1}, \dots, v_{i_j}]$  previous way do
      choose  $v_{i_{j+1}}: c_{i_j i_{j+1}} = \text{Min}_{k \notin \{i_1, \dots, i_j\}} \{c_{i_j k}\}$ 
       $z := z + c_{i_j i_{j+1}}$ 
    end if;
  end;
end;

```

Algorithmus 8.1: Der Pseudocode für den Nearest Neighbour Algorithmus.

Mittels Verwendung eines *heaps* (Haufen), um den nächstgelegenen Punkt zu finden, kann der Algorithmus so modifiziert werden, dass sich eine Komplexität von $\mathcal{O}(n \cdot \log n)$ ergibt. Das bedeutet eine Verbesserung der Laufzeit des Algorithmus. Jedoch gibt es auch hierbei keine Garantie einer Lösung mit optimaler Pfadlänge [HEGELE, 2000].

8.2 Nearest Insertion Algorithmus

Eine weitere Verbesserung des Nearest Neighbour Algorithmus ist der Nearest Insertion (NI) Algorithmus. Dabei wird ein Startpfad, bestehend aus einem Punkt so erweitert, indem einzelne neue Punkte zum Pfad hinzugefügt werden. Dies wird solange wiederholt, bis alle Knoten besucht wurden. Die Strategie besteht darin, den Punkt mit der geringsten Entfernung zu einem Knoten im bereits bestehenden Pfad so einzufügen, dass sich die Pfadlänge um die minimalste Größe erweitert. In dieser Methode ist nicht v_{i+1} der Punkt in T_i , welcher am nächsten zu T_i liegt, sondern bei dem die Entfernung zwischen einem Punkt x und einem Pfad T das Minimum über $y \in T$ von $d(x, y)$ ist (Abbildung 8.2).

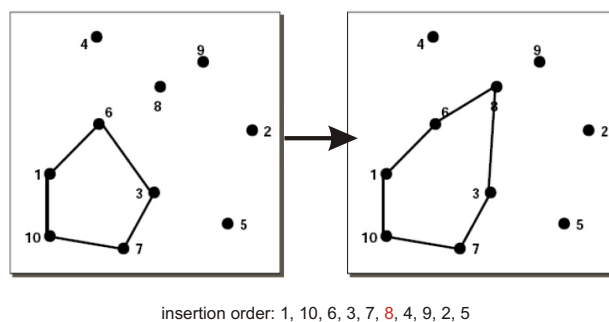


Abbildung 8.2: Beispiel eines aufbauenden Pfades mit Hilfe des Nearest Insertion Algorithmus.

8.3 Rubber Band Algorithmus

Der Rubber Band Algorithmus ist eine Heuristik, die zuerst eine konvexe Hülle über eine Menge von Punkten bildet. Dabei wird ein Teilpfad berechnet, der alle Punkte, die in der äußeren Umgebung der Menge liegen, verbindet. Dann werden die inneren Punkte zum Teilpfad hinzugefügt, indem der jeweils am nächsten zum Pfad liegende Punkt zur iterativ berechneten Pfadlänge hinzugefügt wird. Der Algorithmus terminiert, wenn alle Punkte zum Pfad hinzugefügt wurden.

Die konvexe Hülle kann, zum Beispiel, mit dem Graham Scan Algorithmus berechnet werden. Der Graham Scan Algorithmus arbeitet in zwei Phasen. In *Phase 1*, wird ein einfaches Polygon über die Punkte konstruiert, während in der *zweiten Phase* die konkaven Winkel im Polygon nacheinander gelöscht werden.

In *Phase 1*, wird ein Startpunkt p , zum Beispiel mit der minimalsten y -Koordinate, gewählt. Dann werden die noch vorhandenen Punkte mit Hilfe der Orientierungs- und Vergleichsoperation $less(a, p, q)$ sortiert. So wird ein einfaches Polygon gebildet, wobei die Eckpunkte, die sortierten Punkte darstellen.

In *Phase 2*, werden zwei Datenstrukturen P und H gebildet. P besteht aus einer Liste der Ecken des Polygons. Die Punkte in P werden der Reihe nach passiert. Dann werden die Punkte auf den *stack* (Stapel) H gelegt oder abgeworfen. Nach mehreren Wiederholungen enthält H die Punkte für die konvexe Hülle. Bei der Wiederholung wird der nächste Punkt p von P gewählt, und H wird mit Hilfe einer Aktualisierungsregel auf den neuesten Stand gebracht. Wenn H nicht mehr als zwei Punkte enthält oder wenn für die beiden obersten Punkte q und r des Stacks H gilt, dass

(p, q, r) positiv ist, so wird p auf H gelegt. Andernfalls wird r vom Stack H genommen und die Aktualisierungsregel wird erneut angewendet (Abbildung 8.3).

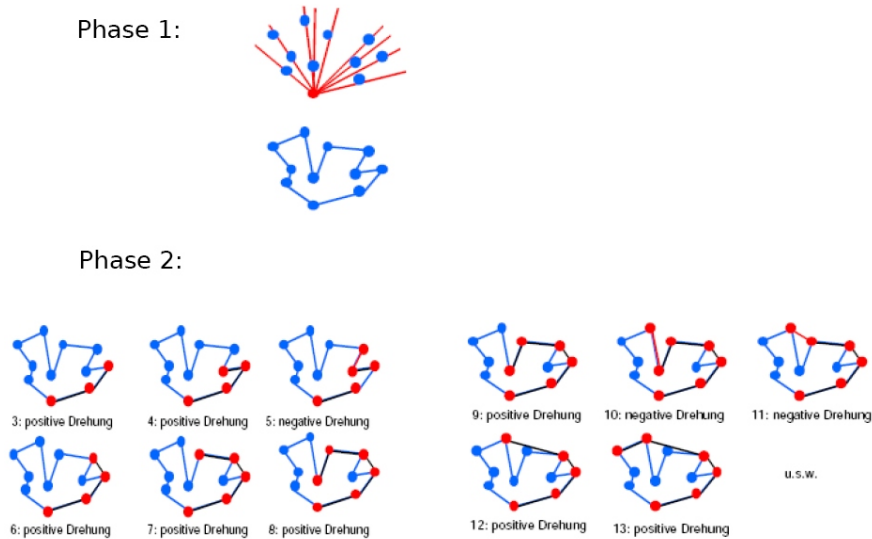


Abbildung 8.3: Beispiel für den Graham Scan Algorithmus.

Die Komplexität des Graham Scan Algorithmus (Algorithmus 8.3) ist im Ganzen $\mathcal{O}(n \cdot \log n)$. *Phase 1* benötigt für das Sortieren der Punkte $\mathcal{O}(n \cdot \log n)$. In *phase 2*, in der jeder Punkt genau einmal auf H gelegt wird und meistens mindestens einmal wieder vom Stapel genommen werden muss, wird $\mathcal{O}(n)$ Zeit benötigt.

Graham-Scan Algorithm

```
Algorithm Scan(S, a): Sequence S of sorted
points; a is an anchor point.
S.insertLast(a)
prev ← S.first()      {prev = a initially}
curr ← S.after(prev)
repeat
  next ← S.after(curr)
  if points (point(prev), point(curr),
point(next)) make a left turn then
    prev ← curr
  else S.remove(curr)
    prev ← S.before(prev)
    curr ← S.after(prev)
until curr = S.last()
S.remove(S.last());
```

Algorithmus 8.3: Der Pseudocode des Graham Scan Algorithmus.

Eine Verbesserung des Algorithmus stellt die Berechnung der Entfernung der inneren Punkte des Polygons dar [SCHLIEDER, 2000].

Der Rubber Band Algorithmus findet in einigen Fällen nicht die optimale Lösung, denn er gibt üblicherweise nur einen einzelnen Pfad zurück.

8.4 Metaheuristiken

Metaheuristiken sind Algorithmen, welche bei kombinatorischen Optimierungsproblemen verwendet werden um gute Lösungen für bestimmte Probleme zu finden. Im allgemeinen ist eine Metaheuristik eine nützliche Methode, welche entwickelt wurde um problem-spezifische Heuristiken in vielversprechende Regionen des Suchraumes zu führen [TANLER, 2004]. Sie kombinieren verschiedene Suchstrategien, wie lokale und globale Suche, Aufbau und Aktualisierung. Die Meta-Algorithmen steuern dabei die eingebetteten Algorithmen. Auch hier gibt es keine Garantie dafür, dass die optimale Lösung gefunden wird und häufig wird nicht angedeutet, wie nahe die optimale Lösung an der gefundenen Lösung liegt. Trotz allem, sind Heuristiken schnell und folglich benötigen sie sehr wenig Rechenschritte.

Der Name Metaheuristik beinhaltet eine Kombination aus zwei griechischen Wörtern. Heuristik stammt von dem Verb *heuriskein* (Griechisch: *εὕρισκειν*, Deutsch: finden) und der Präposition *meta* ab. Bekannte Metaheuristiken sind, zum Beispiel lokale Suche (LS), simulated annealing (SA), tabu search (TS) oder Evolutionäre Algorithmen (EA) (Abbildung 8.4). Im Folgenden Abschnitt soll auf das simulated annealing ein kurzer Blick geworfen werden.

Überblick Metaheuristiken:

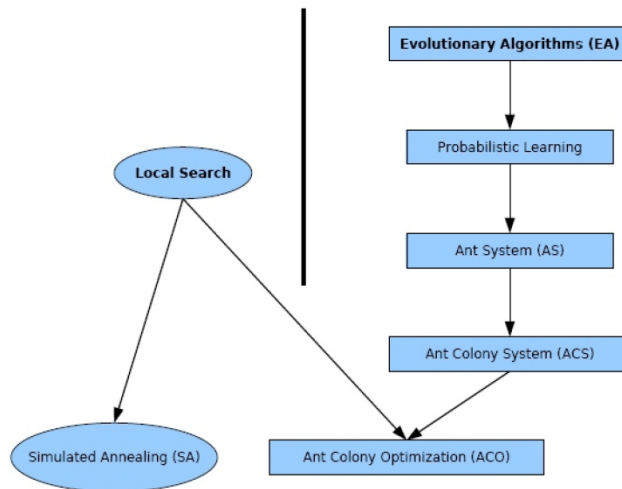


Abbildung 8.4: Ein Überblick auf metaheuristische Verwandtschaften.

8.4.1 Simulated annealing

Simulated annealing (SA) ist ursprünglich ein Prozess aus der Metallurgie. Metalle sind umso härter, je gleichmäßiger ihre Kristallstruktur ist. Beim Abkühlen aus der Schmelze werden an vielen Stellen regelmäßige Strukturen gebildet, an anderen dominieren jedoch chaotische Anordnungen der Atome. Das Ergebnis ist ein Metall, in welchem viele kleine Kristalle ungeordnet nebeneinander liegen, so dass ihre Kristallstruktur nicht optimal verläuft. Deshalb wird das Metall bis kurz unter den Schmelzpunkt erhitzt und dann erneut langsam abgekühlt. Dieser Prozess wird "*tempern* (annealing)" genannt. Somit haben die Atome die Möglichkeit sich neu anzuordnen. Die Atome in den schon gebildeten regelmäßigen Strukturen haben eine niedrigere elektrische Energie als die Atome außerhalb dieser Strukturen.

Das Erhitzen des Metalls muss vorsichtig geschehen, denn wenn es zu schnell erhitzt wird, können alle Atome aus ihren Positionen und regelmäßigen Strukturen gerissen werden. Wird das Metall zu schwach erhitzt, so haben die außerhalb der Kristallstruktur liegenden Atome keine Möglichkeit eine neue Anordnung zu finden. Der Prozess des Temperns wird so lange wiederholt und dabei in jedem Wiederholungsschritt die Temperatur um einige Grad herabgesetzt, bis die Abnahme der Temperatur keine Änderung im Abkühlungsprozess verursacht und den Prozess beendet. Das Ergebnis ist ein hartes und widerstandsfähiges Metall.

Im Jahre 1953 entwickelte W. Metropolis einen Algorithmus für das simulated annealing, den sogenannten Metropolis Algorithmus (Algorithmus 8.4.1). Er ist abhängig von einer Monte-Carlo Simulation¹⁰ des Annealingprozesses und simuliert die Bildung des festen Körpers im Hitzebad. Dabei wird eine Ordnung von Zuständen gebildet, in welchen zuerst ausgehend vom aktuellen Zustand i mit Energie E_i ein nachfolgender Zustand j über eine kleine Störung gebildet wird. Ist

¹⁰ Die **Monte-Carlo Simulation** ist ein Verfahren aus der Stochastik, bei dem sehr häufig durchgeführte Zufallsexperimente die Basis bilden. Es wird versucht, aufgrund der Ergebnisse und mit Hilfe der Wahrscheinlichkeitstheorie analytisch unlösbare Probleme in einem mathematischen Kontext numerisch zu lösen.

$E_i E_j - E_i \leq 0$, dann wird der Zustand j akzeptiert. Ist dies nicht der Fall und es gilt $E_j - E_i > 0$, so wird der Zustand j mit einer Wahrscheinlichkeit von $p = \exp(-\frac{E_j - E_i}{k_B T})$ [k_B : Boltzmann-Konstante, T : Temperatur] akzeptiert. Diese Wahrscheinlichkeit ist das Metropolis-Annahmekriterium für eine aktuelle Lösung (Minimierung) [MERZ, 2002].

```

function simulatedAnnealing(s:S):S
begin
  t = T(0), n = 0; sbest = s;
  repeat
    choice s* ∈ X(s);
    if g(s*, s) > 0
      then s = s*;
    else if exp(g(s*, s)/t) > rand[0, 1)
      then s = s*;
    if g(s, sbest) > 0
      then sbest = s;
    t = T(n);
    n = n+1;
  until n > nmax;
  return sbest;
end;

```

Algorithmus 8.4.1: Der Pseudocode für den simulated annealing Algorithmus.

Somit hatte Metropolis eine Methode zur Lösung von Optimierungsproblemen entwickelt, welche den Weg initiiert, in dem thermodynamische Systeme von einem Zustand in einen anderen wechseln. Diese Methode erfordert nach Metropolis, dass ein System von Teilchen Energiezustände aufweist, in der Weise, in der die thermodynamische Entropie eines gegebenen Temperaturwertes maximiert wird. Dabei sollte der durchschnittliche Energiezustand proportional zu der konstanten Temperatur sein. Diese Methode nannte er simulated annealing (SA) [SCHMIDT].

Metropolis implementierte einen Algorithmus, welcher auch als die "Metropolisregel der Wahrscheinlichkeit" bekannt ist, um das Tempern einer Reihe von Bewegungen darzustellen. Während jeder Bewegung hat das System einige Wahrscheinlichkeitswerte um von seiner aktuellen Konfiguration in eine schlechtere zu wechseln. Die Wahrscheinlichkeit kann über $e^{-\frac{(E_2 - E_1)}{k \cdot T}}$ summiert werden, wobei E_1 die Kosten für die aktuelle Konfiguration darstellt und E_2 die Kosten für die veränderte Konfiguration. Diese Gleichung wird auch das Metropolis Kriterium genannt. Die Fähigkeit in andere Konfigurationen zu wechseln ist das, was SA ermöglicht aus einem lokalen Maxima oder Minima herauszuspringen, während viele andere Algorithmen in diesen Fällen zurückbleiben.

Einige Parameter werden für eine Implementation des SA benötigt. Diese wurden von Davidson und Harel zusammengefasst:

1. Die Menge der Konfigurationen, oder Zustände, des Systems enthalten eine Initialkonfiguration, welche meist zufällig gewählt wird.
2. Eine Erzeugungsregel für neue Konfigurationen, welche normalerweise über die definierte Nachbarschaft jeder Konfiguration gebildet wird und der zufälligen Wahl der folgenden Konfiguration aus der Nachbarschaft der aktuellen Konfiguration.

3. Die Ziel- oder die Kostenfunktion wird über den Konfigurationsraum minimiert, ähnlich zur Energie.
4. Die *cooling schedule* (Zeitplan für die Abkühlung) der Kontrollparameter enthält Initialwerte und Regeln dafür, wann und wie sie verändert werden dürfen. Dies ist ähnlich zur Temperatur, welche in jedem Schritt herabgesetzt wird.
5. Die Beendigungsbedingung, welche gewöhnlich auf der Zeit und auf den Werten der Kostenfunktion basiert, sowie auf den Kontrollparametern.

SA findet nicht immer, und in einigen Fällen gar nicht, die optimale Lösung zu einem gegebenen Problem. Trotz allem, wird SA immer eine bessere Lösung als herkömmliche Heuristiken finden. Wenn das SA auf das TSP angewandt wird, gibt es eine Möglichkeit, dass die Lösung nicht beliebig global ist. Jedoch ist die Lösung normalerweise besser als der gewöhnliche Optimierungsalgorithmus. Der Algorithmus startet mit einer zufälligen Temperatur. Dann wird eine Reihenfolge von Schritten während dieser Temperatur belegt. Ein Schritt beinhaltet die Bildung einer neuen Konfiguration und deren Akzeptierung oder Ablehnung in Hinsicht auf die verwendete Wahrscheinlichkeitsgleichung. Wird ein Schritt akzeptiert, dann wird die Temperatur verändert und die vorherige Prozessabfolge wiederholt. Je höher die Temperatur, desto wahrscheinlicher ist es, dass SA in einen weniger optimalen Zustand übergeht. Nach Schneider, ist diese Wahrscheinlichkeit repräsentiert über die Verwendung der Hamilton-Gleichung, $\rho(\sigma_i \rightarrow \sigma_{i+1}) = \min\{1, \exp(-\frac{\Delta H}{T})\}$, wobei $\Delta H = H(\sigma_{i+1}) - H(\sigma_i)$ die Differenz zwischen den beiden verschiedenen Konfigurationen repräsentiert.

Kirkpatrick, Gelatt und Vecci entdeckten, dass der Prozess aus der Metallurgie zur Lösung des TSP verwendet werden kann und so entstand das simulated annealing für Optimierungsprobleme [TANLER, 2004].

Simulated annealing erlaubt zu jedem Zeitpunkt Aufwärtsbewegungen, so dass es stark auf eine zufällige Anordnung angewiesen ist. Dennoch ist simulated annealing ein grundlegender Suchalgorithmus, bei dem die derzeitige Lösung von Nachbar zu Nachbar wandert, während die Berechnung weiterläuft. Der Schlüsselunterschied zu anderen Ansätzen ist, dass simulated annealing die Nachbarn in einer zufälligen Reihenfolge überprüft, indem es sich zum erstbesten bewegt, welcher gleich gut oder besser in einem Wahrscheinlichkeitstest abschneidet. Ursprünglich wurde von Kirkpatrick et al. [KIRKPATRICK, 1983] und Cerny [CERNY, 1985], der Wahrscheinlichkeitstest verwendet, der von Metropolis et al. [METROPOLIS, 1953] eingeführt wurde, um das physikalische Verhalten von Atomen in einem Hitzebad zu simulieren. Er beinhaltet einen Kontrollparameter *temperature*. Dieser wird beim simulated annealing ständig herabgesetzt, während die Suche fortgesetzt wird. Dies steht in Anlehnung an den physikalischen annealing Prozess.

Das TSP war eines der ersten Probleme, an welches das simulated annealing angepasst wurde. Eigens dazu um als ein Beispiel für Kirkpatrick et al. [KIRKPATRICK, 1983] und Cerny [CERNY, 1985] zu dienen. Seitdem hat das TSP zu einer grundlegenden Testumgebung für Lösungsansätze und deren Varianten geführt. So kann in der Theorie das simulated annealing als ein Optimierungsalgorithmus beschrieben werden.

Wissenschaftler neigen dazu, die oben schon erwähnte *cooling schedule* zu verwenden, die die Temperatur sehr viel schneller herabsetzen kann. Mit so einem Zeitplan, ist das simulated annealing nur ein abschätzender Algorithmus, es wäre jedoch nicht gerechtfertigt mehr für ein NP-hartes Problem, wie das TSP, zu erwarten.

Bei der Anpassung des simulated annealings an das TSP regen Kirkpatrick et al. [KIRKPATRICK, 1983] und Cerny [CERNY, 1985] zur Verwendung einer Nachbarstruktur an, die auf 2-Opt basiert. Dabei werden zwei Kanten gegeneinander ausgetauscht, um die Lösung zu verbessern. Cerny zieht auch eine einfache Bewegung in betracht, in welcher die Positionen zweier Knoten ausgetauscht werden, der Abschnitt dazwischen jedoch unverändert bleibt. Versuche zeigten jedoch, dass dies keinen effektiven Ansatz darstellt.

Leider tritt ein erheblicherer Fehler im obigen Ansatz der Anpassung des simulated annealings an das TSP auf. Die Anzahl der Schritte muss bei jeder Temperatur, welche als die Temperaturlänge

bezeichnet wird, proportional zur Nachbarschaftsgröße sein, wenn eine gute Pfadqualität erzielt werden soll. Dies ist eine Einschränkung für Probleme, wie Graphpartitionierung¹¹, bei denen normalerweise die Nachbarschaftsgrößen $\mathcal{O}(N)$ sind [KIRKPATRICK, 1983]. Für die 2-Opt TSP-Nachbarschaft ist die Größe proportional zu N^2 , so dass dann, wenn die Anzahl der verschiedenen Temperaturen betrachtet wird, die Größe nicht mit N anwächst. Es wird ein Algorithmus erhalten, welcher eine Laufzeit von $\mathcal{O}(N^2)$ und eine feste Proportionalitätskonstante besitzt [JOHNSON, 1995].

Ursprünglich dachte Kirkpatrick daran, das SA auf rechner-basierte Probleme anzuwenden. Er wandte das SA im Jahre 1983 auf verschiedene Optimierungsprobleme an. Von da an, haben viele andere daran gearbeitet und das SA an viele Optimierungsprobleme angepasst. SA ist ein guter Algorithmus, da er sehr allgemein gehalten ist und nicht dazu tendiert in einem lokalen Minimum oder Maximum zurückzubleiben.

Somit lässt sich sagen, dass das simulated annealing einer der besten aktuellen Algorithmen für die Lösung von Optimierungsprobleme ist. Der Algorithmus löst hierbei optimale Probleme, indem er das Abkühlen von Metall in die Kristallstruktur simuliert, dass heißt, können die minimalen und maximalen Punkte auf einem festen Stück Metall bestimmt werden, dann kann man diese auch in Optimierungsproblemen bestimmen. Der ursprüngliche SA Algorithmus arbeitet in Bezug auf das TSP nicht sehr gut. Jedoch kann der Algorithmus so angepasst werden, dass er sehr gute Lösungen für das TSP erzielt. Infolge der Tatsache, dass es keinen allgemeinen SA Algorithmus gibt, der für alle Probleme gute Lösungen erzielt, ist es noch ein weiter Weg in der Erforschung des SA [SCHMIDT].

M. Hauskrecht stellte in seiner Vorlesung einen simulated annealing Ansatz zur Lösung des TSPs vor. Das hier verwendete TSP ist dabei definiert über n Punkte auf einer zweidimensionalen Karte. Die (x, y) Koordinaten definieren die Lage der Punkte. Das Ziel ist, den Pfad zu finden, der die kürzeste Länge aufweist.

Das SA untersucht den Raum aller Pfade, indem es eine zufällige Neuordnung des aktuellen Pfades bildet. Der neue Pfad wird dann akzeptiert, wenn seine Energie (Distanz) kleiner als die Energie des aktuellen Pfades ist. Der Pfad mit einer größeren Energie wird nur mit einer zufällig gewählten Wahrscheinlichkeit $e^{-\frac{\Delta E}{T}}$ noch akzeptiert, wobei ΔE die Energiedifferenz zwischen aktueller und neuer Energie darstellt, und T der Temperaturparameter ist, welcher während der Suche verändert wird.

Der Algorithmus hierzu sollte einige Eigenschaften aufweisen, das heißt,

1. von einem zufälligen Pfad starten und die Energiefunktion des gefundenen Pfades zurückgeben.
2. einen Mutationsmechanismus verwenden, um zufällige Umordnungen des Pfades zu implementieren. Eine Mutation eines Pfades beinhaltet die Drehung der Richtung in der ein Abschnitt des Pfades durchquert wird.
3. eine lineare *cooling schedule* verwenden, die die Temperatur linear in $k + 1$ Schritten herabsetzt, wobei ausgehend von der Anfangstemperatur $T_{initial}$ gestartet und bei der Endtemperatur von 0°C geendet wird.
4. die folgenden Ergebnisse und Statistiken sammeln und ausgeben: (1) Anfangspfad und seine Distanz (Energie), (2) Anfangstemperatur $T_{initial}$, (3) Anzahl der abgelaufenen Pfade, (4) Anzahl der akzeptierten Pfade und (5) der beste gefundene Pfad und seine Distanz (Energie)

Dabei muss noch beachtet werden, dass der annealing Algorithmus eine wahrscheinliche Wahl an Konfigurationen mit höheren Pfadenergien besitzen sollte. Daher sollten noch die folgenden Berechnungen mit einfließen:

¹¹ Bei der **Graphenpartitionierung** wird ein ungerichteter Graph so in Komponenten zerlegt, dass die Anzahl der Knoten aller Komponenten minimal ist. Für allgemeine Graphen ist dieses Problem NP-vollständig.

1. Die Berechnung von $p = e^{-\frac{\Delta E}{T}}$ (wobei $0 \leq p \leq 1$).
2. Die zufällige Wahl einer Zahl x aus dem Intervall $[0, 1]$.
3. Akzeptierung der Wahl, wenn $x \leq p$, sonst Ablehnung der Wahl.

Der Nachteil des SA-Algorithmus ist, dass zu jedem Zeitpunkt nur eine aktuelle Konfiguration beibehalten wird, und dass die Konfigurationen die im nächsten Schritt erhalten werden über "lokale" Veränderungen erzielt werden. So braucht der Algorithmus mehrere Schritte, bis eine gute Konfiguration gefunden wird [HAUSKRECHT, 2005].

Mit simulated annealing kann die Lösung im globalen Minimum gefunden werden. Dies muss jedoch nicht jederzeit geschehen und ist abhängig von der Wahl der verschiedenen Variablen [TANLER, 2004].

8.5 Verwendete Strategien der Versuchspersonen

Um einen Einblick in die Vorgehensweise der Versuchspersonen zu haben, wurden die Versuchspersonen nach dem Experiment befragt, wie sie während der Lösung der Aufgaben vorgegangen sind. Dabei stand vor allem die Aussage im Vordergrund, dass viele der Versuchspersonen sich sowohl im ersten, als auch im zweiten Experiment, an den Regionen orientiert haben. Im ersten Experiment wurde erwähnt, dass zuerst einmal die Symbole auf der *shopping* Liste farblich in die Regionen aufgeteilt wurden und welche dann nacheinander besucht werden konnten. Dies unterstützt die Aussage über eine regionen-basierte Strategie, die zuerst den Pfad grob über die Regionen plant und ihn erst während der Navigation verfeinert. Im zweiten Experiment trat die Aussage auf, dass wie im ersten Experiment, zuerst die Symbole ihren zugehörigen Regionen zugeordnet wurden. Hier jedoch wurde der Pfad über die Symbole gebildet, die die Versuchspersonen am besten im Gedächtnis behalten hatten, da ja hier alle Symbole verdeckt waren. Dies zeigt auch hier eine Tendenz zu einer regionen-basierten Strategie. Mit steigender Anzahl an gelösten Aufgaben war auch ein Anstieg der Qualität der Lösungen zu erkennen (siehe experimentelle Blöcke, Kapitel 5, Abschnitt 5.2). Auch bauten die Versuchspersonen jetzt häufiger einzelne Punkte aus anderen Regionen in den Teilpfad einer Region mit ein, was damit zu einem zwischenzeitlichen Wechsel zwischen den Regionen führte.

Zu beachten ist die Tatsache, dass trotz der vermehrten Aussage für eine Verwendung einer regionen-basierten Strategie, die Versuchsperson nicht immer diese Strategie auch verfolgte.

9 Ansatz eines Modells für das menschliche Verhalten bei der Lösung von TSP-Aufgaben

Im letzten Abschnitt des vorigen Kapitels wurden Strategien der Versuchspersonen diskutiert, die diese zur Lösung der TSP-Aufgabe verwendeten. Dabei viel auf, dass viele der Versuchspersonen eine Strategie verwendeten, die auf den Regionen des Feldes basierten. So eine regionen-basierte Strategie beruht auf der Zuordnung der Punkte eines Pfades in die Regionen, in denen dieser Punkt liegt. Damit kann der Pfad zuerst grob über die Reihenfolge der zu besuchenden Regionen geplant und später, während der Navigation, verfeinert werden, indem die Reihenfolge der Punkte innerhalb einer Region geplant wird. Um diese Strategie in einem Algorithmus zu implementieren, der das Verhalten der Versuchspersonen bei der Lösung solcher TSP-Aufgaben vorhersagen könnte, wurde die Zuordnung der Punkte zu ihren Regionen, sowie die Planung des Pfades in den Regionen in der Verbindung eines Cluster Algorithmus und des Nearest Neighbour Algorithmus implementiert (Abbildung 9). Dabei sollte getestet werden, wie gut sich dieser neue Algorithmus in Bezug auf den reinen Nearest Neighbour Algorithmus und in Bezug auf die tatsächliche Qualität der gefundenen Pfade der Versuchspersonen verhält.

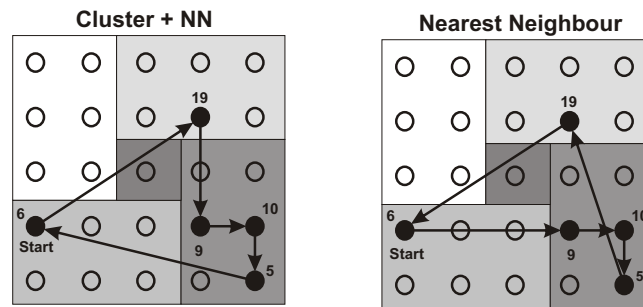


Abbildung 9: Vorhersagen des Cluster + Nearest Neighbour Algorithmus und des Nearest Neighbour Algorithmus.

Beide Algorithmen wurden in R geschrieben und für beide wird eine angepasste Form der im Paket spatstat von R enthaltenen `nnDist.R` verwendet, welche die NN-Distanzen aus einer Menge von Punkten berechnet. In den Algorithmen wird diese Funktion über `evalNNDist.R` aufgerufen.

9.1 Cluster + Nearest Neighbour Algorithmus

Der Aufbau des Cluster + NN Algorithmus, verbindet das Clustern der Punkte eines Pfades über die Regionen und die Berechnung der Reihenfolge der Punkte innerhalb und zwischen den Clusterübergängen mittels Nearest Neighbour Distanzen. Im Großen und Ganzen, werden also zuerst Cluster über die Punkte eines Pfades gebildet. Jeder Cluster enthält dabei die Punkte eines Pfades, die in derselben Region liegen. In Hinsicht auf den hier vorgestellten experimentellen Aufbau, können daher 5 Cluster erhalten werden, wobei 4 der Cluster maximal 6 Punkte und ein Cluster maximal einen Punkt enthalten kann. Nachdem die Cluster eingeteilt wurden, wird der Zentrumspunkt eines Clusters gesucht. Dieser wird dabei über die Formel

$$\text{Koordinaten } (x, y) = \left(\frac{\sum_{i=1}^n x_i}{n}, \frac{\sum_{i=1}^n y_i}{n} \right) \quad (9.1)$$

berechnet. Dann wird der Start-Cluster festgelegt, welcher den Startpunkt des Pfades enthält. Vom Startpunkt ausgehend, wird nun die Distanz zu jedem anderen Cluster berechnet und die minimale Distanz ausgewählt. Das heißt, die Reihenfolge der zu besuchenden Regionen wird über den Start-Cluster und den am nächsten liegenden Cluster bestimmt. Der am nächsten liegende Cluster zum Start-Cluster wird nun zum neuen Start-Cluster und wieder wird der am nächsten liegenden Cluster bestimmt. Dies wird solange wiederholt, bis alle Cluster in einem Array hintereinander gespeichert wurden und die Reihenfolge der zu besuchenden Regionen erhalten wird. Ausgehend von dieser Reihenfolge erfolgt die Berechnung der Reihenfolge der in den Regionen liegenden Punkten, ausgehend vom Startpunkt oder dem letzten Punkt der Reihe im vorangehenden Cluster. Dabei wurde der Nearest Neighbour Algorithmus auf die Pfade und zuvor berechneten Cluster angepasst und in R implementiert.

9.1.1 Implementierung des Cluster + NN Algorithmus

```
#Cluster + Nearest Neighbour Algorithmus

#read in stuff

coords = read.table("coords.txt")
routes = read.table("routes.txt")
regions = read.table("regions.txt")

#settings

targetNo = c(4, 5, 6, 7, 8, 9, 4, 5, ...)
#erstelle Matrix mit NAs gefüllt
matrix = array(NA, dim = c(36, 10))

#wiederhole für jeden Pfad

for(i in 1:nrow(routes)){
  routeDat = routes[i, 2:11]      #Einlesen eines einzelnen Pfads
  targetID = targetNo[routes[i, 1]]
  path = routeDat[, 1:(targetID + 1)]

  for(l in 1:length(path)){
    #für jeden Punkt des Pfades wird die zugehörige Region berechnet
    place = subset(regions, id == path[[l]])
    #berechne regionID
    regionID = place$region
    #berechne Cluster
    if(regionID == 1)      clusterOne = cbind(clusterOne, pointID)
    if(regionID == 2)      clusterTwo = cbind(clusterTwo, pointID)
    if(regionID == 3)      clusterThree = cbind(clusterThree, pointID)
    if(regionID == 4)      clusterFour = cbind(clusterFour, pointID)
    else if(regionID == 5) clusterFive = cbind(clusterFive, pointID)
  }
}
```

```

#Berechne Zentrumsunkte der Cluster mit evalCenterOfCluster.R
centerOfCluster = evalCenterOfCluster(cluster, coords, clusterID)

#Berechne Startpunkt und StartCluster über evalStartCluster.R
startPoint = routeDat[1, 1]
startCluster = evalStartCluster(cluster, startPoint)
#gibt TRUE zurück, wenn der Startpunkt im Cluster liegt

#Array mit neuer Reihenfolge der Cluster
startCluster = sequence[1, ]
newCluster = cbind(newCluster, startCluster)
remainingCluster = sequence[2:nrow(sequence), ]
#Wiederholungen
similes = nrow(sequence) - 1

for(e in 1:similes){
  #wiederhole für jeden restlichen Punkt
  for(n in 1:length(remainingCluster)){
    if(length(remainingCluster) == 1) newCluster = cbind(newCluster,
                                                         remainingCluster)
    else #Berechnung des Paares (startCluster, remainingCluster[n])
      pairs = rbind(startCluster, subset(sequence, id ==
                                         remainingCluster[n, 1]))
      #Aufruf der Funktion evalNNDist.R und Übergabe der Paare
      nnDistPairs = evalNNDist(pairs)
      clusterSequence = rbind(clusterSequence, nnDistPairs)

      calcCluster = min(sequence)
    }
  }

#Berechne restliche Punkte
for(m in 1:length(remainingCluster)){
  if(calcCluster != remainingCluster[1, m]) cluster = cbind(cluster,
                                                           remainingCluster[1, m])
}

#neue restlichen Punkte
remainingCluster = cluster
#neuer Startcluster
startCluster = calcCluster
#neue Clusterreihenfolge
newCluster = cbind(newCluster, calcCluster)
}

#Berechne Punktreihenfolge innerhalb der Cluster über evalNearestPoints.R
start = startPoint
for(t in 1:length(newCluster)){
  if(length(cluster[t]) == 1) newRoute = cbind(newRoute, cluster[t])
  else remainingPoints = cluster
      nearestPoints = evalNearestPoints(remainingPoints, start)
      #berechnen neuen Startpunkt
      start = newRoute[length(newRoute)]
}

#Ausgabe in Matrix
matrix[i, 1:length(path)] = newRoute
}
return(matrix)

```

Die im Cluster + NN Algorithmus verwendete Funktion evalNearestPoint.R ist der auf den Cluster Algorithmus angepasste Nearest Neighbour Algorithmus, der im nächsten Abschnitt beschrieben wird.

9.2 Nearest Neighbour Algorithmus

Der Aufbau des Nearest Neighbour wurde in Kapitel 8 Abschnitt 8.1 behandelt. Der Nearest Neighbour Algorithmus wurde auf die Pfade der Experimente angepasst und in R implementiert.

9.2.1 Implementierung des Nearest Neighbour Algorithmus

```
#Nearest Neighbour Algorithmus

#read in stuff

coords = read.table(„coords.txt“)
routes = read.table(„routes.txt“)

#settings

targetNo = c(4, 5, 6, 7, 8, 9, 4, 5, ...)
#erstelle Matrix mit NAs gefüllt
matrix = array(NA, dim = c(36, 10))

#wiederhole für jeden Pfad

for(i in 1:nrow(routes)){
  routeDat = routes[i, 2:11]      #Einlesen eines einzelnen Pfads
  targetID = targetNo[routes[i, 1]]
  path = routeDat[, 1:(targetID+1)]

  for(l in 1:length(path)){
    #für jeden Punkt des Pfades werden die Koordinaten berechnet
    coordinates = subset(coords, id == path[[l]])
    routeCoords = rbind(routeCoords, coordinates)
  }

  startPoint = routeDat[1, 1]
  #Array mit neuer Reihenfolge der Punkte
  newRoute = cbind(newRoute, startPoint)
  remainingPoints = path[, 2:length(path)]
  #Wiederholungen
  similes = length(path) - 1

  for(e in 1:similes){
    #wiederhole für jeden restlichen Punkt
    for(n in 1:length(remainingPoints)){
      if(length(remainingPoints) == 1) newRoute = cbind(newRoute, remainingPoints)
      else startCoords = subset(coords, id == startPoint)
      #Berechnung des Paares (startPoint, remainingPoints[n])
      pairs = rbind(startCoords, subset(routeCoords, id ==
                                         remainingPoints[1, n]))
      #Aufruf der Funktion evalNNDist und Übergabe der Paare
      nnDistPairs = evalNNDist(pairs)
      sequence = rbind(sequence, nnDistPairs)
    }
    calcRoute = min(sequence)
  }

  #Berechne restliche Punkte
  for(m in 1:length(remainingPoints)){
    if(calcRoute != remainingPoints[1, m]) points = cbind(points, remainingPoints[1, m])
  }

  #neue restlichen Punkte
  remainingPoints = points
  #neuer Startpunkt
  startPoint = calcRoute
  #neue Punktfolgenfolge
  newRoute = cbind(newRoute, calcRoute)
}

#Ausgabe in Matrix
matrix[i, 1:length(path)] = newRoute
}
return(matrix)
```


9.3 Qualität der Algorithmen

In diesem Abschnitt soll die Qualität der oben genannten Algorithmen zum einen gegeneinander, zum anderen gegen die Ergebnisse des ersten Experiments verglichen werden. Hierzu sollen zuerst die Ergebnisse der in diesem Kapitel implementierten Algorithmen miteinander verglichen werden und dann jeder dieser Algorithmen gegen die Ergebnisse des Experiments.

9.3.1 Cluster + NN vs. NN

Die durchschnittliche Fehlerquote lag beim Cluster + NN Algorithmus bei 12,18 %, wo hingegen die des NN Algorithmus bei 8,98 % lag. Auch beim Vergleich der regionalen Gruppe mit der nichtregionalen Gruppe ließen sich deutliche Unterschiede erkennen. Während der Cluster + NN Algorithmus eine Differenz von fast 6 % zwischen der regionalen Gruppe mit 9,28 % und der nichtregionalen Gruppe mit 15,08 % aufwies, war diese Differenz beim NN Algorithmus nicht so stark ausgeprägt. Hier erhielt man eine um knapp 4 % erhöhte Fehlerquote der nichtregionalen Gruppe mit 10,87 % entgegen der regionalen Gruppe mit 7,09 % (Abbildung 9.3.1a). Somit scheint der NN Algorithmus eine leicht bessere Qualität bei den Lösungen aufzuweisen, als der Cluster + NN Algorithmus. Beim Blick auf regionalen und nichtregionalen Gruppen findet sich bei beiden eine niedrigere Fehlerquote bei regionalen Pfaden als bei nichtregionalen Pfaden. In Hinsicht auf die Verwendung einer regionen-basierten Strategie, zeigte sich hier genau die erwartete schlechtere Lösungsqualität der nichtregionalen Gruppe.

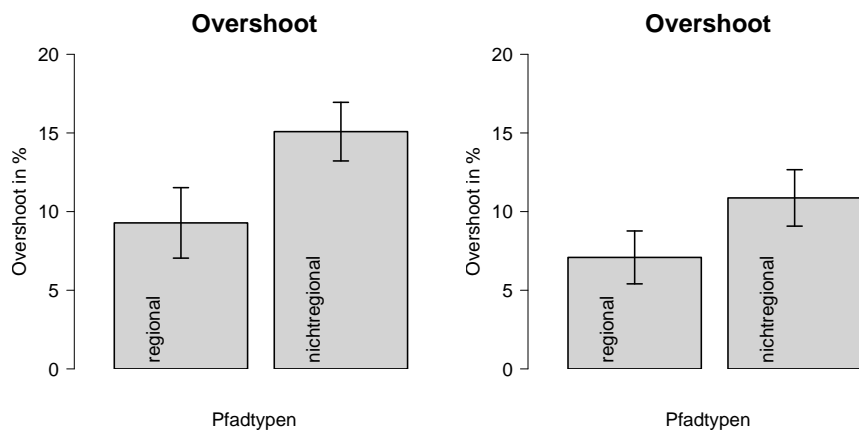


Abbildung 9.3.1a: *Overshoot*.
links: Ergebnis des Cluster + NN Algorithmus.
rechts: Ergebnis des NN Algorithmus.

Die sehr hohe Fehlerquote des Cluster + NN Algorithmus lässt sich damit erklären, dass er, im Gegensatz zum NN Algorithmus auf einer regionen-basierten Strategie aufbaut. Dies hat wiederum zur Folge, dass sich der *Overshoot* genau in der nichtregionalen Pfadgruppe erhöht, da hier mit einer regionen-basierten Strategie kein optimaler Pfad gefunden werden kann und somit jeder Pfad aus dieser Gruppe als suboptimaler Pfad in die Auswertung mit eingeht.

Trotz des Unterschieds von mehr als 3 % in der Fehlerquote, wiesen beide Algorithmen mit durchschnittlich 8,33 % eine geringe Anzahl an optimal gefunden Pfaden auf. Davon lagen 16,67 % in der regionalen Gruppe und 0 % in der nichtregionalen Gruppe (Abbildung 9.3.1b).

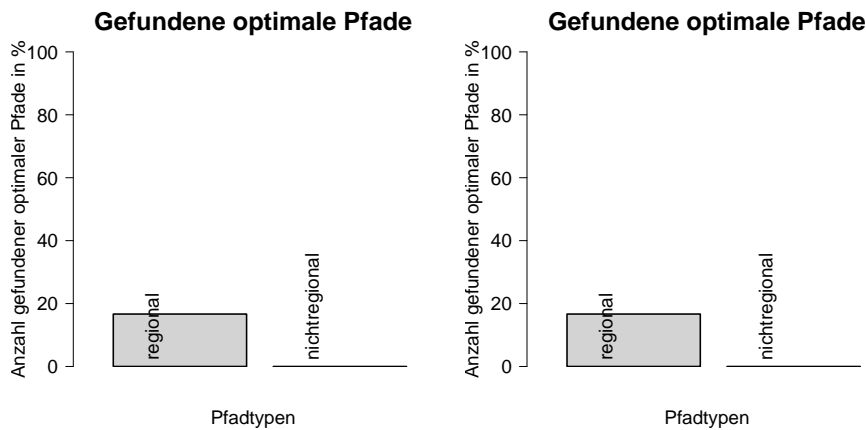


Abbildung 9.3.1b: Durchschnittlich gefunden Anzahl an korrekten Pfaden.
 links: Ergebnis des Cluster + NN Algorithmus.
 rechts: Ergebnis des NN Algorithmus.

Beim Blick auf die optimal gefunden Pfade zeigte sich, beim Cluster + NN Algorithmus die Tendenz, dass bei Pfaden mit 6, 7 und 8 Zielpunkten jeweils an die 17 % korrekt gefundenen Pfaden lag und bei 4, 5 und 9 Zielpunkten keine korrekt gefundenen Pfade vorlagen. Im Unterschied hierzu lag die Tendenz beim NN Algorithmus bei 6 Zielpunkten bei 17 % und bei 8 Zielpunkten bei 34 %. Die restlichen Pfade wiesen keine korrekt gefundenen Pfade auf (Abbildung 9.3.1c). Es scheint hier die Art und der Aufbau des Pfades eine große Rolle zu spielen. Sehr viele Pfade scheinen weder für den einen noch den anderen Algorithmus die Möglichkeit zu bieten, eine optimale Lösung zu finden.

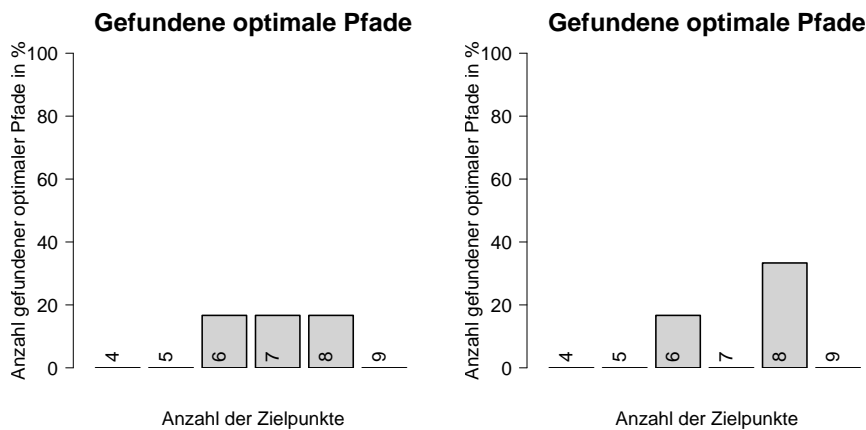


Abbildung 9.3.1c: Durchschnittlich gefunden Anzahl an korrekten Pfaden nach Anzahl der Zielpunkte.
 links: Ergebnis des Cluster + NN Algorithmus.
 rechts: Ergebnis des NN Algorithmus.

Wie erwartet, zeigte sich eine optimale Überschreitung der Regionen in der Gruppe der regionalen Pfade bei den Lösungen des Cluster + NN Algorithmus. Auch wurden die Regionen zu über 80 % optimal in der Gruppe der nichtregionalen Pfade überschritten (Abbildung 9.3.1d). Das lag

daran, dass es nichtregionale Pfade gab, in denen die optimale Regionenüberschreitung mit der Anzahl einer Regionenüberschreitung unter Verwendung einer regionen-basierten Strategie übereinstimmte. Beim reinen NN Algorithmus zeigte sich eine sehr geringere Anzahl an optimalen Regionenüberschreitungen. Dies war zu erwarten, da der Cluster + NN Algorithmus im Gegensatz zum NN auf einer regionen-basierten Strategie aufgebaut ist. Jedoch bedeutet eine optimale Überschreitung der Regionen nicht immer eine optimale Lösung. Dies liegt vorallem daran, dass auch hier die Fehler häufig innerhalb der Regionen auftreten und nicht auf Regionenebene.

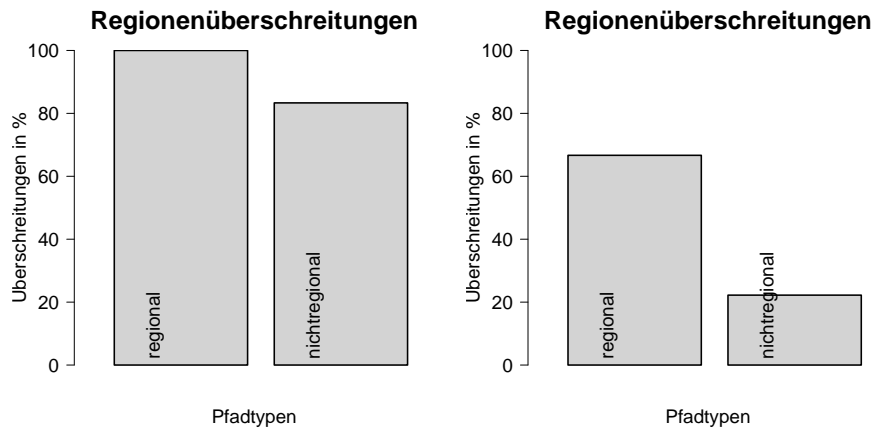


Abbildung 9.3.1d: Durchschnittliche Anzahl an optimalen Regionenüberschreitungen.
 links: Ergebnis des Cluster + NN Algorithmus.
 rechts: Ergebnis des NN Algorithmus.

Wie oben schon erwähnt betrug die optimale Pfadlänge aller 36 Pfade im Durchschnitt eine Länge von 1397,36 cm. Die Pfadlänge der Ergebnisse des Cluster + NN Algorithmus betrug im Schnitt 1569,07 cm, darunter die durchschnittliche Länge der regionalen Pfade mit 1526,39 cm und der nichtregionalen Pfade mit 1611,75 cm. Im Gegensatz hierzu betrug die durchschnittliche Pfadlänge der Ergebnisse des NN Algorithmus 1524,89 cm. Hierbei enthielt die Gruppe der regionalen Pfade eine durchschnittliche Pfadlänge von 1497,36 cm und die nichtregionale Gruppe von 1552,43 cm Länge. Wie oben schon erwähnt, hängt die durchschnittliche Pfadlänge mit der Größe des *Overshoots* zusammen. Ein hoher *Overshoot* resultiert folglich in einer größeren Pfadlänge.

9.3.2 Cluster + NN vs. "Mensch"

Die Ergebnisse des ersten Experiments zeigten eine deutlich bessere Qualität der Lösungen als die des Cluster + NN Algorithmus. Die Qualität des Algorithmus war um circa 7 % schlechter als die der Versuchspersonen. Die Lösungen der Versuchspersonen zeigten mit durchschnittlich 30 % optimal gefundenen Pfaden eine sehr viel höhere Qualität der Lösungen. Vorallem zeigten die Ergebnisse, dass auch in der nichtregionalen Gruppe mit durchschnittlich 22 % die optimalen Pfade gefunden wurden, wohingegen der Algorithmus in dieser Gruppe keine optimale Lösung finden konnte. Durchschnittlich betrug die optimale Pfadlänge aller 36 Pfade eine Länge von 1397,36 cm. Die Pfadlänge der Versuchspersonen betrug im Schnitt 1467,43 cm, darunter die durchschnittliche Länge der regionalen Pfade mit 1460,43 cm und der nichtregionalen Pfade mit 1474,5 cm. Die Pfadlänge der Ergebnisse des Cluster + NN Algorithmus betrug durchschnittlich 1569,07 cm, darunter die durchschnittliche Länge der regionalen Pfade mit 1526,39 cm und der nichtregionalen Pfade mit 1611,75 cm. Dies war jedoch zu erwarten, da der Algorithmus auf einer regionen-basierten Strategie aufgebaut ist und somit in der Gruppe der nichtregionalen

Pfade auch keine optimale Lösung finden konnte. Dies wiederum lässt den Schluss zu, dass nicht alle Versuchspersonen immer die regionen-basierte Strategie anwendeten, sondern vielleicht zwischenzeitlich die Region wechselten, um einen einzelnen Zielpunkt mit in den Pfad aufzunehmen. Der etwas größere *Overshoot*, liegt wohl darin, dass eine Kombination von Cluster Algorithmus und Nearest Neighbour häufiger schlechte Pfade findet und Verbesserungen auf Ebene der Punkte benötigt. Das heißt, eine größere Berücksichtigung der möglichen Anordnung der Punkte innerhalb der Region, sowie die Berücksichtigung von welchem Punkt aus in welche Region gewechselt wird.

9.3.3 NN vs. "Mensch"

Auch in Hinblick auf den Vergleich zwischen NN Algorithmus und den Ergebnissen des ersten Experiments, zeigten sich die gleichen Unterschiede und Probleme. Auch wenn der NN Algorithmus die Regionen nicht berücksichtigt, führte eine NN Strategie nicht immer zu einem optimalen Pfad. Dies wiederum zeigt, dass die Versuchspersonen eine weitreichendere Strategie als die des Nearest Neighbour verwenden. Somit stellt wahrscheinlich die Verbesserung der Kombination zwischen Cluster und NN eine gute Basis dar, eine Vorhersage über das Verhalten der Versuchspersonen zu treffen.

10 Zusammenfassung und Ausblick

In dieser Arbeit wurden zwei Experimente vorgestellt die Wegeplanung am Menschen mit Hilfe des TSPs erforschen. Wer sich schon einmal mit Wegeplanung beschäftigt hat, weiß wie komplex und rechenaufwendig diese Aufgaben sein können. Dies zeigt sich am besten anhand des TSP. Es gibt bisher keinen Algorithmus, der jedes TSP Problem, gleich welcher Größe, in realer Zeit und mit normalem Rechenaufwand lösen kann. Deshalb wird das TSP Problem auch mit der Eigenschaft NP-hart bezeichnet.

Aus einigen bekannten Pfadplanungstrategien am Menschen (Kapitel 1, J. N. MacGregor und T. Ormerod) zeigte sich, dass Menschen anstatt alle möglichen Lösungen zu berechnen, auf Strategien und Heuristiken vertrauen müssen. Aus einer früheren Studie zur Pfadplanung am Menschen kristallisierte sich die Anwendung einer regionen-basierten Strategie heraus. Dieser Ansatz wurde in den vorigen Kapiteln vorgestellten Experimenten aufgegriffen. Dabei zeigte sich, dass Menschen zum einen sehr gute und annähernd optimale Lösungen der TSP-Aufgaben erzeugen, ohne dabei ungewöhnlich viel Zeit zur Berechnung dieser Lösungen zu investieren. Es zeigte sich auch, dass mit einer reinen regionen-basierten Strategie kein vollständig zufriedenstellendes Modell zur Vorhersage des Verhaltens von Menschen bei der Pfadplanung beschrieben werden kann. Eine regionen-basierte Strategie wurde zwar sehr häufig verwendet, das heißt, es wurde zuerst ein grober Pfad über die Regionen gebildet, jedoch blieb ungeklärt, wie die Versuchspersonen innerhalb der Regionen agieren. Hier scheint jede Versuchsperson eine eigene Strategie zu verfolgen.

Desweiteren wurde im zweiten Experiment untersucht, wie sich das Ortsgedächtnis auf die Strategie bei der Wegeplanung auswirkt. Es erscheint den Versuchspersonen leichter zu fallen, sich zuerst die Regionen einzuprägen und über die Projektion der Regionen eine Möglichkeit zu finden, die genauen Lokalisationen der Punkte in der Umgebung einzuordnen. Hier spielte jedoch das Gedächtnis eine wesentliche Rolle, in wie weit die Versuchspersonen sich das Feld einprägen konnten. Auch hier führte dies zur Verwendung einer regionen-basierten Strategie, aber der Unsicherheitsfaktor, wo sich ein Zielpunkt in der experimentellen Umgebung befindet, führte oft zur Vernachlässigung dieser Strategie und dazu, dass zuerst ein grober Pfad über die Punkte gebildet wurde, die die Versuchspersonen sicher aus dem Langzeitgedächtnis abrufen konnten. Da jede Person in dieser Hinsicht eine andere Art des Lernens zeigt, kann keine Voraussage über eine bestimmte Lernstrategie getroffen werden.

Da noch sehr wenig Arbeit in die Entwicklung eines Modells zur Vorhersage des Verhaltens von Menschen bei der Pfadplanung investiert wurde, war ein weiteres Ziel dieser Arbeit, einen Algorithmus zu implementieren, der diese Vorhersagen unterstützt. Hierzu wurde ein Algorithmus basierend auf der regionen-basierten Strategie implementiert, der Cluster + NN Algorithmus. Hier werden Punkte einer Region zu Clustern zusammengefasst und über die Berechnung der Nearest Neighbour Distanzen eine mögliche Reihenfolge der zu besuchenden Punkte ausgegeben. Wie sich auch im Falle der Experimente zeigte, führt eine regionen-basierte Strategie nicht immer zu einer optimalen Lösung. Auf Basis des Cluster + NN Algorithmus führte dies deshalb nur in etwa 16 % der Fälle zu einer korrekten Lösung. Deshalb sollte die Berechnung einer Reihenfolge innerhalb der Punkte mit einer anderen Strategie betrachtet werden, anstatt mit der Nearest Neighbour Strategie, was hier eindeutig keine optimale Wahl zu sein scheint. Das heißt, die Verwendung einer anderen Strategie plus die regionen-basierte Strategie, führt vielleicht zu einem annähernd besseren Modell zur Pfadplanung am Menschen.

Zusammenfassend lässt sich sagen, dass mit einer steigenden Anzahl an Zielpunkten, Planungsstrategien angewendet werden, um die Belastung des Arbeitsgedächtnisses, sowie den Rechenaufwand zu reduzieren. Dabei sind hierarchische Planungsentwürfe, zum Beispiel eine regionen-basierte oder Cluster-Strategie, einleuchtende Hilfsmittel um diesen Rechenaufwand und die Belastung des Arbeitsgedächtnisses zu reduzieren. Cluster- und regionen-basierte Strategien repräsentieren somit einfache Pfadplanungsheuristiken um diese Reduktion zu gewährleisten. Dies gilt sowohl für das Memorieren der Zielpunkte während der Pfadplanung als auch für den vollständig geplanten Pfad.

Des Weiteren lässt sich sagen, dass es viele reelle Anwendungsmöglichkeiten des TSP gibt, die

modifiziert sind und nicht nur bedingungslos der klassische Definition folgen. Meist sind Modifikationen der Formulierungen essentiell um eine effektive Anpassung an das zu lösende Problem zu gewährleisten. Dabei gibt es einen fließenden Übergang zwischen den Variationen, welche auf das klassische TSP reduziert werden können. Andere müssen jedoch als Erweiterungen angesehen werden und schließlich gibt es noch die Spezialfälle des TSPs und dessen verwandte Probleme. Die Berücksichtigung solcher Variationen in Zusammenhang mit den Anwendungsmöglichkeiten des TSP ist von besonderem Interesse, denn viele der Algorithmen, die zur Lösung des klassischen TSP entwickelt wurden, können mittels Modifikation zur Lösung aller TSP-Arten herangezogen werden.

11 Literatur

- [AMANN] G. Amann, R. Wippinger. *abenteuer psyche*. 1. Auflage, Abbildung 3.16
- [BADDELEY, 1974] Alan D. Baddeley, G. Hitch. *Working memory*. In G. A. Bower (Ed.), *Recent advances in learning and motivation*, Vol.8, 1974. New York: Academic Press.
- [BADDELEY, 1975] Alan D. Baddeley, N. Thomson and M. Buchanan. *Word length and the structure of short-term memory*. *Journal of Verbal Learning and Verbal Behavior*, **14**, pp. 375-589, 1975
- [BADDELEY, 1984] Alan D. Baddeley, V. J. Lewis and G. Vallar. *Exploring the articulatory loop*. *Quarterly Journal of Experimental Psychology*, **36**, pp. 233-252, 1984
- [BADDELEY, 1986] Alan D. Baddeley. *Working memory*. Oxford: Oxford University Press, 1986
- [BADDELEY, 1992] Alan D. Baddeley. *Working memory*. *Science*, **255**, pp. 556-559, 1992
- [BADDELEY/LOGIE, 1992] Alan. D. Baddeley and R. H. Logie. *Auditory imagery and working memory*. In D. Reisberg (Ed.), *Auditory imagery*, pp. 179-197, Hillsdale 1992, NJ: Erlbaum.
- [BADDELEY, 1996] Alan D. Baddeley. *Exploring the central executive*. *The Quarterly Journal of Experimental Psychology*, 1996, **29A(1)**, pp. 5-28
- [BADDELEY, 1997] Alan D. Baddeley. *Human Memory: Theory and Practice*. (Revised Addition). UK: Psychology Press, 1997
- [BADDELEY, 1998] Alan D. Baddeley, S. E. Gathercole and C. Papagno. *The phonological loop as a language learning device*. *Psychological Review*, **105**, pp. 158-173, 1998
- [BADDELEY, 1999] Alan D. Baddeley and R. H. Logie. *Working memory: The multiple-component model*. In A. Miyake & P. Shah (Eds.), *Models of working memory*, pp. 28-61, New York 1999: Cambridge University Press.
- [BADDELEY, 2000] Alan D. Baddeley. *The episodic buffer: a new component of working memory?* *Trends in Cognitive Sciences*, **4(11)**, pp. 417-423
- [BADDELEY, A. D., 2000] Alan D. Baddeley. *The phonological loop and the irrelevant speech effect: Some comments on Neath*. *Psychonomic Bulletin and Review*, **7**, pp. 544-549, 2000
- [BADDELEY, 2001] Alan D. Baddeley. *Is Working Memory Still Working?* *American Psychologist*, Vol. 56, No. 11, pp. 851-864, 2001
- [BADDELEY, A. D., 2001] Alan D. Baddeley. *Comment on Cowan: The magic number and the episodic buffer*. *Behavioral and Brain Sciences*, **24**, pp. 117-118, 2001
- [BADDELEY, 2002] Alan D. Baddeley. *Is working memory still working?* *European Psychologist*, Vol. 7, No. 2. June 2002, pp. 85-97
- [BADDELEY, 2003] Alan D. Baddeley. *Working memory: looking back and looking forward*. *Nature reviews, Neurocience*, Vol. 4, October 2003.
www.nature.com/reviews/neuro
- [BRAVER, 2000] T. S. Braver, J. D. Cohen. *On the control of control: The role of dopamine in regulating prefrontal function and working memory*. In S. Monsell & J. Driver (Eds.), *Control of cognitive processes: Attention and performance XVIII*, pp. 713-737, Cambridge, 2000, MA: MIT Press.
- [BUNGE, 2000] Silvia A. Bunge, Torkel Klingberg, Richard B. Jacobson and John D. E. Gabrieli. *A resource model of the natural basis of executive working memory*. Neurosciences Program. Stanford University Medical Center, Stanford, CA 94305-5404, and Department of Psychology, Stanford University, Stanford, CA 94305-2130. Communicated by P. S. Goldman-Rakic, Yale University

- School of Medicine, New Haven, CT, December 30, 1999 (received for review June 10, 1999). PNAS, March 28, 2000, Vol. 97, No. 7, 3573-3578. Bunge et al. www.pnas.org
- [BURKHART] Helmar Burkhart. *Lektion 8: Geometrische Algorithmen*. Depart. Informatik, Universität Basel. Helmar.Burkhart@uni.bas.ch
- [CERNY, 1985] V. Cerny. *A Thermodynamical Approach to the Travelling Salesman Problem: An Efficient Simulation Algorithm*. J. Optimization Theory and Appl. **45**, 1985, pp. 41-51
- [COHEN, 1996] J. D. Cohen, T. S. Braver & R. C. O'Reilly. *A computational approach to prefrontal cortex, cognitive control, and schizoPhilosophicalphrenia: Recent developments and current challenges*. Philosophical Transactions of the Royal Society (London) B, **351**, pp. 1515-1527, 1996
- [COOPER] Don Cooper. *The development of ant systems to solve dynamic travelling salesman problems*.
- [GRUBER, 2003/2004] W. Gruber. *Brain Modelling: Physikalische Modelle für das Gehirn auf zellulärer Basis*. Kapitel: *Das Gedächtnis*, S. 102-115, Universität Wien, Institut für Experimentalphysik, WS/SS 2003/2004, Skript: http://brain.exp.univie.ac.at/yunterlagen/SS_2.pdf
- [GUAN] Ying-Hua Guan. *Reexamining the modality effect from the perspective of Baddeley's working memory model*. Graduate Program: Task-oriented Communication. Department of Linguistics and Literary Science, University of Bielefeld, P. O. Box: 10 01 31, 33 501 Bielefeld, Germany. yinghua.guan@uni-bielefeld.de
- [HAUSKRECHT, 2005] Professor Milos Hauskrecht. *Problem assignment 3*. University of Pittsburgh, CS 2710 Introduction to Artificial Intelligence. Handout 3. Wednesday, September 28, 2005
- [HEGELE, 2000] Mario Hegele. *Entwicklung einer Simulationsumgebung für kontextbezogenen Informationszugriff in einer mobilen Umgebung*. Studienarbeit Nr. 1790, pp. 18-20, 18. 12. 2000, Universität Stuttgart. Fakultät für Informatik, Institut für parallele und verteilte Höchstleistungsrechnersysteme, Breitwiesenstraße 20-22, D-70565 Stuttgart.
- [HOCHREITER, 1997] S. Hochreiter & J. Schmidhuber. *Long short term memory*. Neural Computation, **9**, pp. 1735-1780, 1997
- [INGBER, 1993] Lester Ingber. *Simulated Annealing: Practice versus theory*. Lester Ingber Research, P.O.B. 857, McLean, VA 22101, November 18, 1993, pp. 29-57. ingber@alumni.caltech.edu
- [JOHNSON, 1995] David S. Johnson, Lyle A. McGeoch. *The Traveling Salesman Problem: A case study in local optimization*. Johnson: Room C239, AT&T Labs, Florham Park, NJ 07932, dsj@research.att.com, McGeoch: Department of Mathematics and Computer Science, Amherst College, Amherst, MA 01002, lam@cs.amherst.edu, November 20, 1995
- [JOHNSON-LAIRD, 1983] P. N. Johnson-Laird. *Mental models*. Cambridge 1983, England: Cambridge University Press.
- [KIRKPATRICK, 1983] S.Kirkpatrick, C. D. Gelatt, JR and M. P. Vecchi. *Optimization by Simulated Annealing*. Science **220**, 13 May 1983, pp. 671-680
- [KÖHLER/MEINEL] Jan Köhler, Bernd Meinel. *Grundlagen der Informationsbearbeitung*. Technische Universität Chemnitz.
- [LIN, 1973] S. Lin, B. W. Kernighan. *An effective Heuristic Algorithm for the TSP*. Operations Research, **21**, pp. 498-516, 1973
- [MACGREGOR, 1996] J. N. MacGregor, T. Ormerod. *Human performance on the traveling salesman problem*. Perception & Psychophysics. 1996, **58** (4), pp. 527 - 539
- [MAGUIRE, 2000] E. Maguire et. al. *Navigation-related structural change in the hippocampi of taxi drivers*. Proceedings of the National Academy of Science (PNAS). April 11, 2000, vol. 97, no. 8, pp. 4398 - 4403

- [MENZEL] Randolph Menzel. FU Berlin.
www.neurobiologie.fu-berlin/WieGehirneLernen.pdf
- [MERZ, 2002] Dr. Peter Merz. *Moderne heuristische Optimierungsverfahren: Meta-Heuristiken*. Wilhelm-Schickard-Institut für Informatik - WSI-RA, Sand 1, Raum A 316, Universität Tübingen. Vorlesung SS 2002.
pmerz@informatik.uni-tuebingen.de .
- [METROPOLIS, 1953] W. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller and E. Teller. *Equation of state calculations by fast computing machines*. J. Chem. Psych. **21**, pp. 1087-1092, 1953
- [MURRAY, 1968] D. J. Murray. *Articulation and acoustic confusability in short-term memory*. Journal of Experimental Psychology, **78**, pp. 679-684, 1968
- [O'REILLY] Randall C. O'Reilly & Michael J. Frank. *Making Working Memory Work: A Computational Model of Learning in the Prefrontal Cortex and Basal Ganglia*. Department of Psychology, University of Colorado Boulder, 345 UCB, Boulder, CO 80309, oreilly@psych.colorado.edu, frankmj@psych.colorado.edu. Neural Computation, in press: Supported by ONR grants N00014-1-0246 and N00014-1-0428, and NIH grants MH64445 and MH069597
- [O'REILLY, 1999] R. C. O'Reilly, T. S. Braver & J. D. Cohen. *A biologically based computational model of working memory*. In A. Miyake & P. Shah (Eds.), *Models of working memory: Mechanism of active maintenance and executive control*, pp. 375-411, New York, 1999: Cambridge University Press.
- [O'REILLY, 2000] R. C. O'Reilly & Y. Munakata. *Computational explorations cognitive neuroscience: Understanding the main by simulating the brain*. Cambridge, MA, 2000: MIT Press.
- [REINELT, 1994] G. Reinelt. *The Traveling Salesman: Computational Solutions for TSP Applications*. **840**, Lecture Notes in Computer Science, Springer Verlag 1994: Berlin.
- [SCHLIEDER, 2000] Christoph Schlieder. *Praktische Informatik II: Geometrische Algorithmen*. Vorlesung 9, pp. 4-5, SS 2000, Universität Bremen.
- [SCHMIDT] Nate Schmidt. *Simulated Annealing*.
- [SCHMITTING, 1999] Walter Schmitting. *Das Traveling-Salesman-Problem: Anwendungen und heuristische Nutzung von Voronoi-/Delauny-Strukturen zur Lösung euklidischer, zweidimensionaler Traveling-Salesman-Probleme*. Zugl.: Düsseldorf Univ. Diss., 1999
- [SCHRIJVER] Alexander Schrijver. *On the history of combinatorial optimization (till 1960)*. CWI and University of Amsterdam. Mailing address: CWI, Kruislaan 413, 1098 SJ Amsterdam, The Netherlands.
- [SQUIRE, 1992] L. R. Squire. *Declarative and nondeclarative memory: Multiple brain systems supporting learning and memory*. Journal of Cognitive Neuroscience, **4**, pp. 232 - 243
- [SWOBODA, 2004] Hannah Swoboda. *Lernen und Gedächtnis*. SS 2004
- [TANLER, 2004] Martin Tanler. *Das "Traveling Salesman Problem (TSP)" als Beispiel für Optimierungsprobleme*. Dipl.-Math. Christian Vogt, 17. Mai 2004
- [ZENKOVA/RIEGER, 2004/2005] Diana Zenkova, Carolin Rieger. *Gedächtnis: Enkodierung und Speicherung*. Seminar Lernen, WS 2004/2005, JLU Gießen.

Anhang

Dieses Kapitel führt die oben erwähnten, jedoch nicht gezeigten Abbildungen verschiedener Kapitel an. Im Anhang A werden Abbildungen des ersten Experiments gezeigt und in Anhang B Abbildungen des zweiten Experiments. In Anhang C sind einige Fotos der Experimente zu sehen.

A Experiment 1

Im Anhang A liegen die Abbildungen zu Kapitel 5 bei.

A.1 Überschreitungen der Regionengrenzen

Die Anzahl an optimalen Regionenüberschreitungen wurde für jeden Pfad berechnet. Das heißt, es wurde ausgehend von der optimalen Lösung gezählt, wie häufig zwischen den Regionen gewechselt wurde. Bei regionalen Pfaden konnten mehr oder gleich viele Regionenüberschreitungen auftreten, da die Pfade so entworfen wurden, dass weniger Überschreitungen dazu führten, dass nicht alle Zielpunkte besucht wurden. Mehr Überschreitungen deuten hier darauf hin, dass Zielpunkte vergessen wurden und diese noch nachträglich in den Pfad miteingebaut werden mussten. Bei nichtregionalen Pfaden konnten mehr, gleich viele oder weniger Regionenüberschreitungen auftreten. Werden weniger Regionengrenzen überschritten als bei der optimalen Lösung berechnet, so wurde eine regionen-basierte Strategie verwendet. Die nichtregionalen Pfade wurden so konstruiert, dass bei Verwendung einer regionen-basierten Strategie, weniger oder gleich viele Regionengrenzen überschritten wurden und dem zu Folge zu einem suboptimalen Pfad führten. Bei einer optimalen Lösung hätten die Versuchspersonen die Regionen häufiger wechseln müssen, um einzelne Punkte in anderen Regionen miteinzubeziehen. Das Auftreten von mehr Regionenüberschreitungen führte bei nichtregionalen Pfaden zu der Annahme, dass die Regionengrenzen häufiger als in der optimalen Lösung vorgesehen überschritten wurden, beispielsweise wenn ein Zielpunkt vergessen wurde. Somit zeigen die Abbildungen A.1a und A.1b, dass auch im Falle von nichtregionalen Pfaden, die regionen-basierte Strategie verwendet wurde und dies in der Überschreitung weniger Regionengrenzen, als in der optimalen Lösung vorgesehen, resultierte.

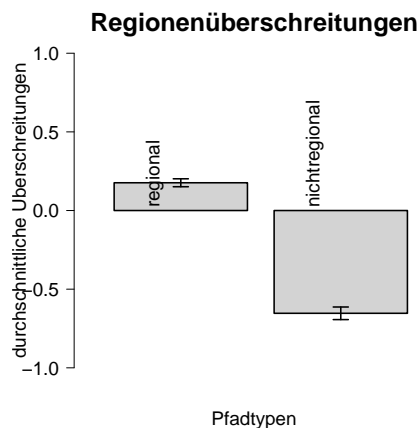


Abbildung A.1a: Anzahl der Regionenüberschreitungen bei beiden Pfadtypen.

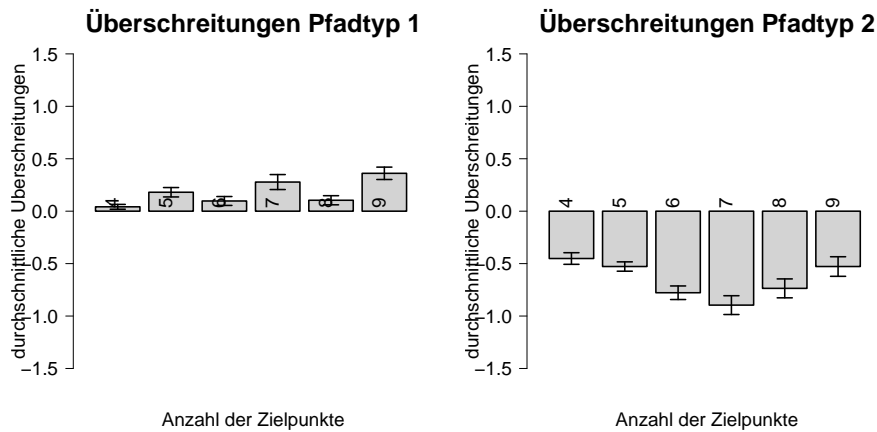


Abbildung A.1b: Anzahl der Regionenüberschreitungen bei beiden Pfadtypen nach Anzahl der Zielpunkte.
links: regionale Pfadgruppe.
rechts: nichtregionale Pfadgruppe.

A.2 Regionen-basierte Strategie

Wie in Kapitel 5, Abschnitt 5.5 dargestellt, kann anhand der Regionenüberschreitungen berechnet werden, wie häufig eine regionen-basierte Strategie verwendet wurde. Dabei wurde zuerst für die optimale Lösung der 36 Pfade berechnet, wie häufig die Versuchsperson die Regionen wechseln musste, falls sie eine regionen-basierte Strategie verwendet hatte. Das heißt, Region für Region besuchte. Dann wurde anhand der Pfade der Versuchspersonen die Anzahl an überquerten Regionengrenzen berechnet und mit der Anzahl an Regionenüberschreitungen der optimalen Lösung verglichen. In der nachfolgenden Tabelle A.2A sind die Unterschiede zwischen männlichen und weiblichen Versuchspersonen dargestellt.

Geschlecht	regional	nichtregional
männlich	\bar{O} 0,54 mal	\bar{O} 0,23 mal
weiblich	\bar{O} 0,56 mal	\bar{O} 0,22 mal

Tabelle A.2A: Durchschnittliche Anwendung der regionen-basierten Strategie bei männlichen und weiblichen Versuchspersonen.

B Experiment 2

In Anhang B liegen Abbildungen zu Kapitel 6 bei.

B.1 Korrekte Pfade

In der zweiten Auswertung zeigte sich das gleiche Bild wie in der ersten Auswertung (Kapitel 6, Abschnitt 6.2.3). Die Anzahl an korrekt gefundenen Pfaden lag mit 26 % in der regionalen

Gruppe und mit 12 % in der nichtregionalen Gruppe, nur knapp unterhalb der in Abschnitt 6.2.3 erhaltenen Ergebnisse (Abbildung B.1). Dies lag daran, dass der Prozentsatz aus durchschnittlich mehr Pfaden berechnet wurde als dies in der Auswertung ohne fehlerhafte Pfade der Fall war. Das heißt, die Höhe der Anzahl gefundener optimaler Pfade war in beiden Auswertungsarten die Gleiche, nur der Prozentsatz war in der zweiten Auswertung geringer, da dieser aus einer größeren Menge an Pfaden berechnet wurde.

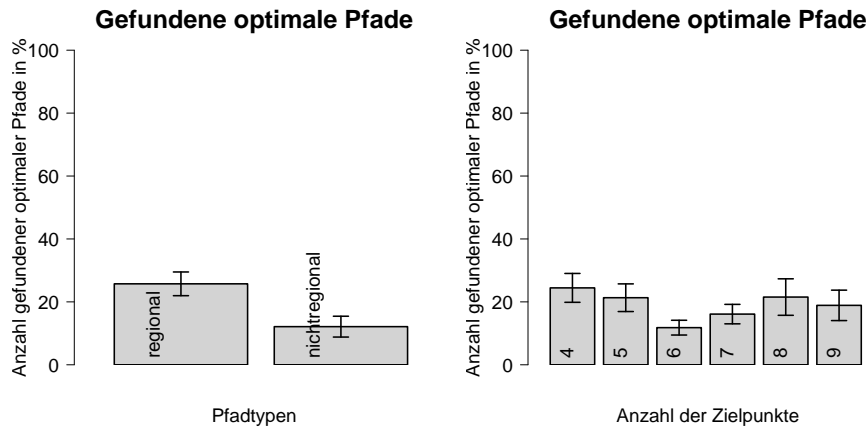


Abbildung B.1: Anzahl der korrekt gefundenen Pfade zwischen den Pfadtypen und nach Anzahl der Zielpunkte.

Es tritt eine starke Signifikanz zwischen regionalen und nichtregionalen Pfaden auf (Welch Two Sample t-test: $t = 2,71$, $df = 45,247$, $p = 0,009$; paired t-test: $t = 5,55$, $df = 23$, $p = 1,201 \cdot 10^{-5}$). Somit hatten die Pfadtypen einfluss auf die Anzahl der korrekt gefundenen Pfade. Auch hier zeigte sich ein deutlicher Unterschied zwischen regionalen und nichtregionalen Regionen. Beim Blick auf die korrekt gefundenen Pfade in Abhängigkeit von der Anzahl der Zielpunkte, ließ sich nur feststellen, dass bei Pfaden mit nur vier Zielpunkten mehr als 20 % der optimalen Pfade von den Versuchspersonen gefunden wurden, bei den restlichen Zielpunkten ließ sich jedoch fast keine Tendenz feststellen (Pearson's product-moment correlation: $p < 0,001$, $r = 0,28$) (Abbildung 5.3 rechts). Dies steht im Gegensatz zu der im Vorfeld getroffenen Annahme, dass es schwerer werden sollte die optimalen Pfad zu finden, je mehr die Anzahl der Zielpunkte ansteigt.

Wie in der ersten Auswertung (Kapitel 6, Abschnitt 6.2.3) ließ sich zwischen weiblichen und männlichen Versuchspersonen die gleiche Tendenz erkennen (Tabelle B.1A).

Geschlecht	insgesamt	regional	nichtregional
männlich	Ø 24,30 %	Ø 30,85 %	Ø 17,68 %
weiblich	Ø 15,16 %	Ø 22,08 %	Ø 8,16 %

Tabelle B.1A: Anzahl der korrekt gefundenen Pfade bei weiblichen und männlichen Versuchspersonen.

B.2 Überschreitungen der Regionengrenzen

Wie in Abschnitt A.1 schon erwähnt, wurde in der nichtregionalen Pfadgruppe weniger zwischen den Regionen gewechselt, als dies bei einer optimalen Lösung der Fall sein sollte. Somit sieht man

auch hier, dass bei nichtregionalen Pfaden häufig eine regionen-basierte Strategie verwendet wurde. In den folgenden Abbildungen B.2a und B.2b sind beide Auswertungen des zweiten Experiments dargestellt.

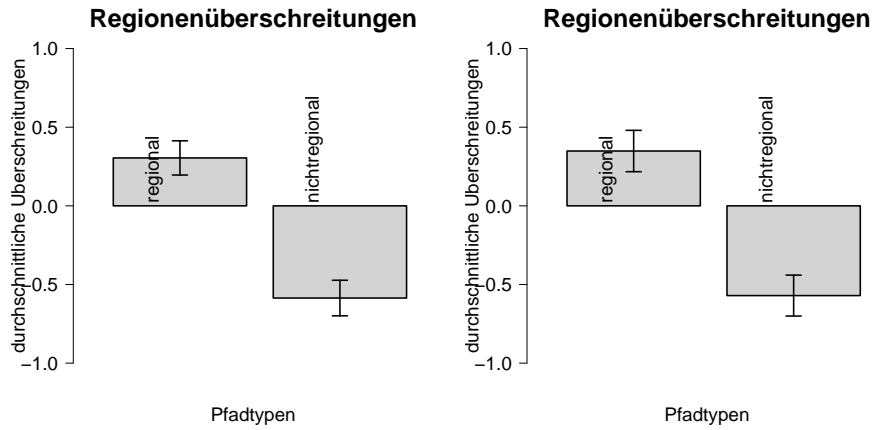


Abbildung B.2a: Regionenüberschreitungen.
links: erste Auswertung.
rechts: zweite Auswertung.

Der t-Test der zweiten Auswertung zeigte eine Signifikanz zwischen regionalen und nichtregionalen Pfaden auf (Welch Two Sample t-test: $t = 2,23$, $df = 46$, $p = 0,03$; paired t-test: $t = 11,43$, $df = 23$, $p = 5,766^{-11}$). Die Pfadtypen hatten einfluss auf die Anzahl der Regionenüberschreitungen.

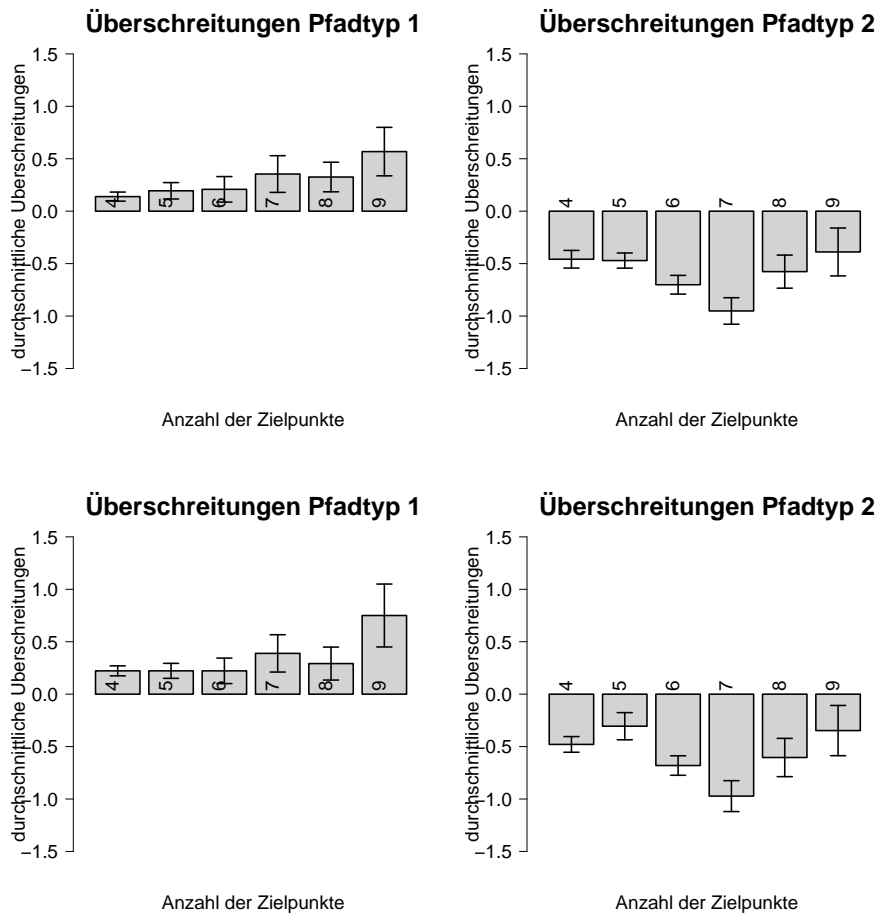


Abbildung B.2b: Regionenüberschreitungen bei den beiden Pfadtypen.
 oben: erste Auswertung, regionale Pfadgruppe (links), nichtregionale Pfadgruppe (rechts).
 unten: zweite Auswertung, regionale Pfadgruppe (links), nichtregionale Pfadgruppe (rechts).

Eine ANOVA wies in der ersten Auswertung einen starken signifikanten Effekt zwischen den Zielpunkten auf ($F(5,115) = 5,127$; $p = 0,003$; $\eta^2 = 0,359$), sowie eine ANOVA in der zweiten Auswertung ($F(5,115) = 6,081$; $p = 0,001$; $\eta^2 = 0,209$; $MSE = 0,370$). Die Anzahl der Zielpunkte schien somit einfluss auf die Anzahl der Regionenüberschreitungen zu haben.

B.3 Regionen-basierte Strategie

Wie im ersten Experiment schon einmal erwähnt (Kapitel 5, Abschnitt 5.5) konnte anhand der Regionenüberschreitungen berechnet werden, wie häufig eine regionen-basierte Strategie verwendet wurde. Dabei wurde zuerst für die optimale Lösung der 36 Pfade berechnet, wie häufig die Versuchsperson die Regionen wechseln musste, falls sie eine regionen-basierte Strategie verwendet hatte, das heißt, Region für Region besuchte. Dann wurde anhand der Pfade der Versuchspersonen die Anzahl an überquerten Regionengrenzen berechnet und mit der Anzahl an Regionenüberschreitungen der optimalen Lösung verglichen. In der nachfolgenden Tabelle B.3A sind die Unterschiede zwischen männlichen und weiblichen Versuchspersonen dargestellt, aufgeteilt in die beiden Auswertungen des zweiten Experiments.

Auswertung	Geschlecht	regional	nichtregional
(1)	männlich	Ø 0,58 mal	Ø 0,22 mal
(1)	weiblich	Ø 0,42 mal	Ø 0,10 mal
(2)	männlich	Ø 0,47 mal	Ø 0,19 mal
(2)	weiblich	Ø 0,38 mal	Ø 0,08 mal

Tabelle B.3A: Häufigkeit der verwendeten Regionenstrategie bei weiblichen und männlichen Versuchspersonen in beiden Auswertungen.

C Fotos

Hier finden sich Fotos des Versuchsaufbaus und einer Versuchsperson beim Lösen der TSP-Aufgaben.

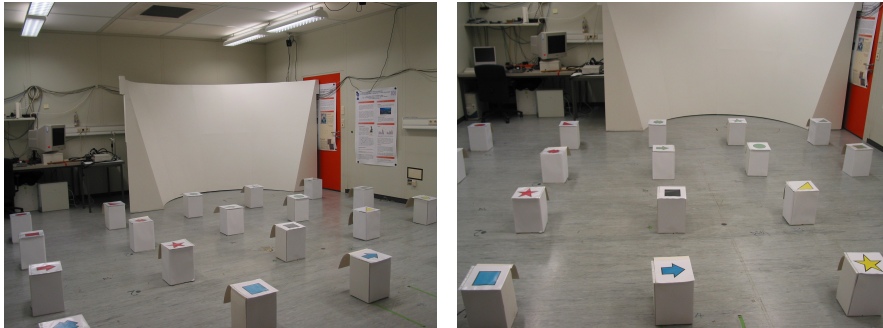


Foto C1: Aufbau der experimentellen Umgebung.



Foto C2: Versuchsperson bei der Navigation.

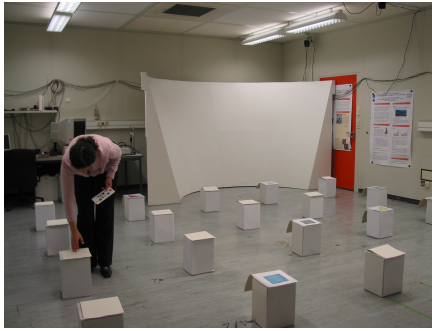


Foto C3: Versuchsperson beim Lösen einer TSP-Aufgabe.