

EIN NETZWERKANALYTISCHER ANSATZ ZUR  
UNTERSUCHUNG DER KOMPLEXITÄT DES  
RUSH-HOUR-SPIELS



EIN NETZWERKANALYTISCHER ANSATZ ZUR UNTERSUCHUNG DER  
KOMPLEXITÄT DES RUSH-HOUR-SPIELS

MAREIKE BOCKHOLT

Prof. Dr. Gerhard Reinelt  
Prof. Dr. Katharina Zweig

Institut für Informatik  
Ruprecht-Karls-Universität

04. März 2013

Mareike Bockholt: *Ein netzwerkanalytischer Ansatz zur Untersuchung der Komplexität des Rush-Hour-Spiels* (Bachelorarbeit), 04. März 2013  
mareike@mathphys.fsk.uni-heidelberg.de  
Matrikelnummer: 2820920



---

## ABSTRACT

---

In cognitive psychology, human problem solving has been an active field of research over the past decades. During this period, methods and possibilities in computer sciences and network analysis have been greatly refined. Looking at psychology's challenging questions from a computer scientist's point of view proves to be worthwhile for both disciplines. Our network analytic approach to the analysis of a game's problemspace – a widely accepted concept in the psychological community – exploits the huge potential of computer-based methods. This enables us to develop complexity measures for the popular board game *Rush Hour*, which we review by using algorithmic approaches to understand the producer's difficulty classifications. Furthermore, we conduct and analyze a study involving 97 subjects. Based on the findings, we investigate complexity measures, the subjects's navigation through the problem space as well as the game's perceived complexity. On the one hand, the results of the analysis suggest a more finely nuanced grading by difficulty for the *Rush Hour* levels. On the other hand, they contribute to a better understanding of the human perception of complexity.

---

## ZUSAMMENFASSUNG

---

Menschliches Problemlösen stellt seit Jahrzehnten ein aktives Forschungsgebiet in der kognitiven Psychologie dar. Dies ist ein Zeitraum, in dem sich die Methoden und Möglichkeiten der Informatik und Netzwerkanalyse enorm weiterentwickelt haben. Der von informatischen Prinzipien geprägte Blick auf psychologisch motivierte Fragestellungen erweist sich als ein lohnender Ansatz, aus dem beide Disziplinen Erkenntnisse ziehen können. Wir nutzen das Potential computergestützter Analysen, indem wir netzwerkanalytische Methoden auf den Problemraum eines Spiels anwenden, ein anerkanntes Konzept in der kognitiven Psychologie. Dies erlaubt uns, Komplexitätsmaße für das beliebte Brettspiel *Rush Hour* zu entwickeln, die wir durch algorithmische Methoden und anhand der herstellerseitigen Schwierigkeitsangaben bewerten können. Zudem stellen wir eine Studie vor, deren Konzeption auf dieser Bewertung basiert. Wir verwenden deren Ergebnisse, um die Komplexitätsmaße, die Navigation der Versuchspersonen durch den Problemraum sowie die empfundene Komplexität der Spiele zu analysieren. Die Resultate dieser Analyse legen einerseits eine verfeinerte Kategorisierung von *Rush Hour* Spielen in Schwierigkeitsgrade nahe, andererseits leisten sie einen Beitrag zum allgemeinen Verständnis dessen, was Menschen als komplex empfinden.



---

## DANKSAGUNG

---

KATHARINA ZWEIG für die netzwerkanalytische und algorithmische Ausbildung, die diese Arbeit überhaupt erst möglich gemacht hat.

GERHARD REINELT für seine Unterstützung bei dieser interuniversitären Bachelorarbeit mit einem Thema, das ich sonst nicht hätte wählen können.

HORVÁT EMŐKE-ÁGNES und ANDREAS SPITZ für ihren unverzichtbaren und unermüdlichen Rat und Beistand.

THINKFUN, dem Hersteller der *Rush-Hour*-Spiele, für die Genehmigung und die Unterstützung dieses Projekts.



---

## INHALTSVERZEICHNIS

---

0	EINLEITUNG	1
0.1	Motivation . . . . .	1
0.2	Struktur der Arbeit . . . . .	2
1	KOMPLEXITÄT	3
1.1	Komplexität in der Informatik . . . . .	3
1.2	Komplexität in der Psychologie . . . . .	4
2	DAS SPIEL RUSH HOUR	9
2.1	Das Spiel . . . . .	9
2.2	Definitionen . . . . .	9
3	UNTERSUCHUNG DER LEVELKARTEN	13
3.1	Daten . . . . .	13
3.2	Komplexitätsmaße . . . . .	14
3.3	Implementierung . . . . .	17
3.4	Ergebnisse . . . . .	22
3.4.1	Verteilungsanalyse . . . . .	22
3.4.2	Korrelationen der Maße . . . . .	25
3.4.3	Clustering der Spiele anhand ihrer Komplexitätsmaße . . . . .	27
4	EXPERIMENT	31
4.1	Konzeption: Aufbau und Durchführung des Experiments . . . . .	31
4.2	Versuchsergebnisse . . . . .	33
4.2.1	Spieler . . . . .	35
4.2.2	Abgebrochene und gelöste Level . . . . .	36
4.2.3	Lösungen der Spieler . . . . .	38
4.2.4	Korrelation zwischen Maßen und Versuchsdaten . . . . .	39
4.2.5	Irrwege im Zustandsraum . . . . .	41
4.2.6	Analyse der Zustände . . . . .	43
4.2.7	Unintuitive Züge . . . . .	48
5	ZUSAMMENFASSUNG UND AUSBLICK	55
5.1	Zusammenfassung der Ergebnisse . . . . .	55
5.2	Ausblick . . . . .	56
5.3	Relevanz der Resultate . . . . .	57
	LITERATURVERZEICHNIS	59





---

## EINLEITUNG

---

### 0.1 MOTIVATION

Taglich werden Menschen mit Problemen konfrontiert, deren Losung sie (in Abhangigkeit von ihrer Schwierigkeit) mehr oder weniger Anstrengung kostet. Einige nehmen sie gar nicht mehr als Probleme wahr, von anderen fuhlen sie sich berfordert: sei es die Aufgabe, von der Arbeit nach Hause zu gelangen ohne in einen Stau zu geraten, ihren Tagesablauf zu planen oder ein Schachspiel zu gewinnen. Gleichzeitig stehen uns heutzutage gigantische technische Moglichkeiten zur Verfugung, Aufgaben anzugehen, die vor ein paar Jahrzehnten nicht einmal denkbar waren. Vor allem in Situationen, in denen groe Mengen an Informationen verarbeitet oder eine fur Menschen berwaltigende Menge an Abhangigkeiten bercksichtigt werden mssen, besitzen Rechner unschlagbare Vorteile in der Geschwindigkeit der Ausfuhrung von Basisoperationen, der Parallelisierung von Prozessen und der strukturierten Speicherung von Daten. Umgekehrt gibt es klassische Beispiele fur aus menschlicher Sicht einfache Probleme, die mithilfe eines Rechners nicht mit vergleichbar geringem Aufwand gelost werden konnen, z. B. in der Bildverarbeitung.

Doch was zeichnet Probleme aus, die einen Menschen herausfordern, jedoch von Computern innerhalb weniger Rechenschritte gelost werden konnen? Ist es allein die Anzahl an Schritten oder die Menge an gleichzeitig verarbeitenden Informationen? Kann man vielleicht sogar ein oder mehrere Kriterien finden, um Probleme nach ihrer Komplexitat fur Menschen zu klassifizieren? Uns interessiert in diesem Zusammenhang besonders die Frage, ob es moglich ist, objektive Mae zu finden, die eine Vorhersage ber die von Menschen empfundene Komplexitat erlauben. Vor der Versuchung *die* Komplexitat eines Systems messen zu wollen, warnte bereits Dorner 1989 [5]. Doch was passiert, wenn man ein Problem betrachtet, das weder dynamisch noch intransparent noch vernetzt ist, sondern im Gegenteil vorhersehbar, bestimmbar und leicht zu berblicken ist – ein Problem also, das die meisten der von Dorner eingefuhrten Eigenschaften komplexer Systeme nicht besitzt? Was erhalt man weiterhin, wenn man ausschlielich die Eigenschaften *eines* Problems betrachtet und auf diesen intrinsischen Eigenschaften des speziellen Problems aufbauend Methoden zur Beurteilung seiner Komplexitat entwickelt?

In dieser Arbeit stellen wir einen Ansatz vor, um am Beispiel eines verbreiteten Brettspiels ein solches Problem in verschiedenen Auspragungen darauf zu untersuchen, wie schwer dessen Losbarkeit von Menschen empfunden wird. Durch den quantifizierbaren und von informatischen Prinzipien gepragten Blick auf die psy-

chologisch motivierte Fragestellung erlangen wir eine neue Perspektive, die ohne diesen Aspekt bisher verborgen geblieben ist. Wir untersuchen die Möglichkeiten, mit netzwerkanalytischen Methoden, die wir auf den Problemraum anwenden, neue Erkenntnisse zu erwerben.

## 0.2 STRUKTUR DER ARBEIT

Wir beschäftigen uns in dieser Arbeit mit dem Spiel *Rush Hour* und untersuchen seine Komplexität im informatischen und psychologischen Sinne. Unser Ziel ist es, mithilfe von Methoden aus Informatik und Netzwerkanalyse, einen Katalog an Kriterien zu erstellen, die die von Menschen empfundene Schwierigkeit eines *Rush-Hour*-Spiels bestmöglich vorhersagen können. Dazu gehen wir in zwei Schritten vor: zunächst stellen wir mögliche Komplexitätsmaße vor und vergleichen diese mit den Schwierigkeitseinteilungen des Hersteller des Spiels. Danach konzipieren wir mit Hilfe dieser Ergebnisse ein eigenes Experiment, das unsere Komplexitätseinschätzung anhand der Daten unserer Versuchspersonen bewertet.

Mit diesem Ziel vor Augen strukturieren wir die vorliegende Arbeit folgendermaßen: in Kapitel 1 geben wir einen kurzen Überblick, was in der Psychologie und was in der Informatik unter Komplexität verstanden wird. Zudem geben wir hier eine knappe Einführung, was bereits über das menschliche Problemlösen bekannt ist. In Kapitel 2 beschreiben wir das Spiel *Rush Hour* zunächst informell, bevor wir seine Eigenschaften formal definieren und einige weitere nützliche Definitionen vorstellen.

In Kapitel 3 widmen wir uns der theoretischen Komplexität des *Rush-Hour*-Spiels. Dazu stellen wir zunächst einige Komplexitätsmaße vor und berechnen diese für alle Spielkonfigurationen, die der Hersteller bisher publiziert hat. Diese Ergebnisse vergleichen wir mit der Schwierigkeitseinschätzung des Herstellers, um unsere Maße zu bewerten. In diesem Kapitel beschreiben wir außerdem kurz die benötigte Implementierung für die Berechnung dieser Maße.

In Kapitel 4 beschäftigen wir uns mit der Konzeption, der Durchführung und den Ergebnissen unseres Experiments, in dessen Rahmen unsere Versuchspersonen einige ausgewählte *Rush-Hour*-Level lösen. Wir beschreiben zunächst den Aufbau und die Rahmenbedingungen der Studie, bevor wir auf die Kriterien hinter unserer Levelauswahl eingehen. Als Hauptteil dieser Arbeit stellen wir die Ergebnisse des Experiments vor und vergleichen diese mit unserem Verständnis von Komplexität.

Abschließend geben wir eine kurze Zusammenfassung über unsere Resultate und legen dar, in welcher Form die Ansätze dieser Arbeit in Zukunft weiterentwickelt werden könnten.



---

## KOMPLEXITÄT

---

Der Begriff der Komplexität ist mit vielen verschiedenen Bedeutungen belegt, je nachdem in welcher Wissenschaft oder in welchem Zusammenhang er benutzt wird. Im Alltag wird der Begriff meist ohne eine präzise Definition verwendet, in Wissenschaften wie der Mathematik, Informatik, Physik oder Psychologie ist der Begriff zwar definiert, bezeichnet jedoch jeweils unterschiedliche Konzepte. Im Folgenden beschäftigen wir uns mit dem Begriff Komplexität wie er in der Informatik und in der Psychologie verwendet wird, und stellen die Bedeutung des Begriffs in diesen beiden Wissenschaften vor.

### 1.1 KOMPLEXITÄT IN DER INFORMATIK

Dem Begriff ist in beiden Disziplinen gemeinsam, dass er sich in gewisser Weise mit der Schwierigkeit von Problemen und ihrer Lösung beschäftigt. Die zentrale Frage in der Informatik lautet hierbei: wie schnell kann für ein gegebenes Problem eine Lösung gefunden werden? Abhängig von der Antwort auf diese Frage werden Probleme dann in Komplexitätsklassen eingeteilt.

Um diese Frage jedoch stellen und beantworten zu können, müssen wir einige Begriffe und Modelle einführen. In der Informatik betrachten wir nur Probleme, die so formuliert werden können, dass sie durch eine Abfolge von Instruktionen gelöst werden können (zum Beispiel von einem Computer). Es spielt dabei prinzipiell keine Rolle, ob diese Instruktionenfolge von einem Hochleistungscomputer oder von einem Menschen mit Papier und Stift ausgeführt wird. Aus dieser Überlegung heraus wird jedoch klar, dass ein Modell benötigt wird, das fähig ist, das Lösen eines Problems darzustellen. Dieses Modell einer Rechenmaschine soll es erlauben sein, die Komplexität eines Problems zu bemessen, indem es die Ressourcen betrachtet, die benötigt werden, um das Problem zu lösen. Als ein solches Modell kann die nach Alan Turing [17] benannte Turingmaschine dienen. Diese besteht im simpelsten Fall aus einem unendlich langen Band mit unendlich vielen Feldern, in denen jeweils ein Zeichen aus einem endlichen Alphabet stehen kann, sowie einem Schreiblesekopf, der das Zeichen eines Feldes lesen oder ein Zeichen in ein Feld schreiben kann. In jedem Schritt liest die Turingmaschine nun das Zeichen in dem Feld mit der Kopfposition, schreibt in dieses Feld ein Zeichen und bewegt den Schreiblesekopf um eine Zelle nach rechts oder links. Zu einer Turingmaschine gehört außerdem eine Menge an Zuständen, in denen sie sich befinden kann und – das eigentliche Herzstück – ein Programm, das aufgrund des aktuellen Zustands und des gelesenen Zeichens festlegt, welche Aktion die Turingmaschine ausführt und in welchen Zustand sie übergeht.

Mithilfe dieses Modells kann nun die Komplexität eines Problems präzise analysiert werden. Wir können die Anzahl der Schritte, die eine Turingmaschine bräuchte, um das Problem zu lösen, als Maß für die Komplexität des Problems verwenden. Alternativ dazu kann auch die Anzahl der Felder, die die Maschine während der Lösung benötigt, betrachtet werden. Wichtig ist allerdings zu erwähnen, dass beide Maße Funktionen in Abhängigkeit der Eingabelänge sind.

Wir nennen ein Problem dann zum Beispiel *in quadratischer Laufzeit lösbar*, wenn es eine Turingmaschine gibt, sodass die Anzahl der Rechenschritte, die sie benötigt, um die Lösung des Problems zu liefern, asymptotisch nicht schneller als quadratisch in der Eingabelänge wachsen. Dieses Modell erlaubt es also, ein objektives Maß der Komplexität eines Problems anzugeben.

## 1.2 KOMPLEXITÄT IN DER PSYCHOLOGIE

Die Psychologie auf der anderen Seite beschäftigt sich mit der menschlichen Herangehensweise an das Problemlösen. Diese scheint nicht so einfach zu fassen zu sein und noch schwerer quantifizierbar. Hier stellen sich Lesern mit informatischem Hintergrund sofort zwei Fragen: Was heißt hier *komplex* und mit welcher Bedeutung wird der Begriff *Problemlösen* verwendet? Bei unserer Beschäftigung mit der psychologischen Betrachtungsweise der Begriffe *Komplexität* und *Problemlösen* ist uns zunächst aufgefallen, dass innerhalb der Psychologie eine ganze Reihe von sehr unterschiedlichen Definitionsvorschlägen existieren, jedoch eine allgemeingültige Definition, auf die man sich geeinigt hat, nicht existiert.

Anderson [3] definiert *Problemlösen* als „zielgerichtetes Verhalten, das oft das Aufstellen von Teilzielen beinhaltet, um die Anwendung von Operatoren zu ermöglichen“. Er führt drei wesentliche Merkmale des Problemlösens ein: Zielgerichtetheit, Zerlegung in Teilziele und Anwendung von Operatoren. Hierbei verwendet er den Terminus *Operator* als eine Handlung, die einen gegebenen Problemzustand in einen anderen überführt.

Andere Autoren liefern weitere mehr und weniger präzise Definitionsvorschläge, den meisten jedoch ist gemeinsam, dass sie zielorientiertes Denken und Handeln beinhalten, um eine Situation aus bestimmten Gründen mithilfe einer Sequenz von Handlungen in eine andere umzuwandeln, wobei in der Ausgangssituation nicht unmittelbar klar ist, welche Sequenz von Handlungen dies leistet.

Für den Begriff *Komplexität* existieren vermutlich genauso viele Definitionsvorschläge wie für *Problemlösen*, jedoch sind die Ansätze hier noch weiter gefächert und schwieriger auf einen Nenner zu bringen. Quesada et al. [14] wagten 2001 den Versuch, den oft verwendeten Begriff des *complex problem solving* zu definieren, konnten aber keine einzige Eigenschaft herausstellen, die sowohl hinreichend als auch notwendig ist, um ein Problem *komplex* zu nennen. Dörner [5] spricht von der Komplexität eines Systems oder einer Situation, wenn es beziehungsweise sie folgende Eigenschaften hat:

- Komplexität: Anzahl der Merkmale, die das System beschreiben
- Vernetztheit: Abhängigkeit der Merkmale untereinander
- Intransparenz: nicht alle Merkmale des Systems sind zugänglich
- Dynamik: das System entwickelt sich auch ohne eigenes Handeln weiter
- Polytelie: mehrere und sich womöglich widersprechende Zielsetzungen.

Funke [8] weist jedoch auch darauf hin, dass es

[...] keineswegs so [ist], dass alle Eigenschaften eines komplexen Problems *spezifisch* für diese Art von Problemen wären, d. h., nur bei diesen auftreten können. Nebenwirkungen infolge von Vernetztheit gibt es beispielsweise auch bei einfachen Puzzle-Problemen [...]. Die einzelnen Subziele können nicht unabhängig voneinander bearbeitet werden, da die mit einem Zug verbundenen Nebenwirkungen unvermeidlich sind und mitbedacht werden müssen. Vernetztheit ist damit trotz seiner enormen Bedeutung für diese Problemklasse kein Alleinstellungsmerkmal von komplexen Problemen.

Dörner führt weiter an, dass es keine objektive Komplexität eines Systems geben kann, sondern nur eine subjektive, da die Komplexität eines Systems stark davon abhängt, wie es vom Einzelnen wahrgenommen wird. Ein wesentlicher Faktor hier wird sicherlich sein, inwieweit die Person mit dem vorliegenden System vertraut ist, sodass sie die von Miller [11] eingeführten *bits of information* über das System zu sogenannten *chunks* gruppieren und inhaltlich ordnen kann, also viele einzelne Informationsstückchen zu sinnvollen Gruppen oder Mustern zusammenfassen kann, die dann als Ganzes wiedererkannt werden können.

Dies kann jedoch noch nicht die ganze Wahrheit sein, denn es gibt eine Unzahl von komplex (im Sinne von schwierig) erscheinenden Situationen, die diese Merkmale nicht aufweisen: nehmen wir unser im Kapitel 2 eingeführtes Spiel *Rush Hour*, so fällt uns auf, dass dieses weder viele beschreibende Merkmale noch eine Eigendynamik besitzt. Zudem besitzt es genau ein Ziel und dem Spieler sind im Voraus alle Informationen über das System bekannt. Jedoch ist die Idee der Vernetztheit zutreffend, wenn gegenseitige Blockaden der Autos als Abhängigkeiten bezeichnet werden, und es ist insofern dynamisch, als dass sich das System zwar nicht von selbst, aber doch durch Zutun des Problemlösers ändert. Einige Autoren, wie zum Beispiel Wirth und Klieme [18], bezeichnen Systeme dann bereits als dynamisch. Damit besitzt *Rush Hour* die nach Funke [7] beiden wichtigsten Eigenschaften eines komplexen Systems: Vernetztheit und Dynamik.

Zu erwähnen ist hier auch, dass im Regelfall in der kognitiven Psychologie die Komplexität eines Problems nicht gleichzusetzen ist mit der empfundenen Schwierigkeit eines Problems. Kotovsky et al. [19] zeigten bereits 1985, dass es große Unterschiede im Lösungserfolg der Versuchspersonen von demselben Problem gibt – beispielsweise abhängig davon, in welcher Weise das Problem den Versuchspersonen präsentiert wird, was für eine *cover story* verwendet wird, um das Problem darzustellen, oder inwiefern der Rahmen des Problems konsistent mit den Erfahrungen der Versuchsperson ist.

Wirth und Klieme [18] fassen die für sie wichtigsten Teile eines Lösungsprozesses eines Problems folgendermaßen zusammen:

1. Suche, Strukturierung, Repräsentation und Integration von Informationen
2. Schlussfolgerndes Denken (*reasoning*) auf Basis eines Modells
3. Planung der Handlungen und Schritte im Lösungsprozess
4. Ausführung und Evaluation der Handlungen
5. ständige Verarbeitung von äußeren Informationen und Feedback
6. Optimierung durch Lernen und Anwendung von Strategien

Die Wichtigkeit der genannten Komponenten und der Aufwand, der diese Komponente zur Lösung eines Problems erfordert, erlauben ihnen eine Klassifikation von Problemen in grob zwei Gruppen. In Problemen wie *Rush Hour* oder dem viel untersuchten Spiel *Tower of Hanoi* muss kaum Aufwand in die Suche nach Informationen oder Entwicklung eines geeigneten Modells gesteckt werden, doch die Planung und Ausführung einer Folge von Aktionen, die den Anfangszustand in den gewünschten Zustand bringt, stellt die Hauptaufgabe dar. Punkt (5) und (6) seien nur wichtig, wenn die geplante Aktionsfolge nicht erfolgreich ist [18].

Die andere Gruppe enthält den in einigen Studien untersuchten Typ von Problemen wie der *Lohhausen-Simulation* (siehe zum Beispiel Dörner [5]), die in verschiedenen Ausprägungen alle von ihm eingeführten Eigenschaften von komplexen Systemen besitzt. Bei diesem Typ von komplexen Problemen sind die Schritte (5) und (6) besonders wichtig, damit die Versuchsperson fähig ist, eine Lösung zu finden.

Ein Ansatz, die Komplexität eines Problems zu klassifizieren, stellt die von Halford et al. [9] eingeführte relationale Komplexität dar. Sie stellen die Idee einer Relation vor, die die Elemente enthält, die parallel verarbeitet werden müssen um ein Problem zu lösen. Die Anzahl der miteinander in Verbindung stehenden Elemente, die *Dimension* der Relation, schlagen sie als Komplexität des Problems vor. Laut Halford et al. können Menschen nur Relationen der Dimension vier oder niedriger verarbeiten, ansonsten würden die Relationen in kleinere aufgeteilt und nacheinander verarbeitet. Die Anzahl der Elemente, die gleichzeitig verarbeitet werden müssen (und nicht aufgeteilt und seriell verarbeitet werden können), wäre somit ein Maß für die Komplexität eines Problems.

Eine weitere Annäherung an den Begriff der Komplexität kann die Untersuchung sein, wie Menschen für solche Situationen Lösungen finden. Auch wenn wir kein mathematisches Modell wie das der Turingmaschine für das menschliche Problemlösen vorstellen können, gibt es doch einige bereits bekannte Ansätze, die Menschen benutzen, um ein gegebenes Problem zu lösen, die im Folgenden vorgestellt werden.

Alan Newell hat bereits 1979 das Konzept des Problemraums vorgestellt [12] als Tupel einer Menge von Problemzuständen und einer Menge von Operatoren, so dass jeder Operator einen Zustand als Eingabe erhält und einen neuen Problemzustand ausgibt. Sequentielle Anwendung von Operatoren definiert dann einen Pfad durch den Problemraum. Das zu einem Problemraum gehörende Problem kann dann als Menge von Anfangszuständen, einer Menge von Zielzuständen und einer Menge von Pfadbeschränkungen angesehen werden. Das Problemlösen besteht dann darin, einen Pfad von einem Anfangszustand zu einem Endzustand zu finden, ohne die Pfadbeschränkungen zu verletzen.

Das Lösen eines Problems kann damit also als das Absuchen seines Problemraums bezeichnet werden. Unmittelbar stellt sich die Frage, welche Methoden Personen anwenden, um den Problemraum zu durchsuchen. Dazu müssen sie in jedem Problemzustand entscheiden, welchen der ihnen zur Verfügung stehenden Operatoren sie wählen. Im Laufe der letzten Jahrzehnte – auch als Ergebnisse von Forschungsarbeiten im Bereich der Künstlichen Intelligenz – wurden mehrere Methoden bzw. Heuristiken identifiziert, nach denen Menschen bei der Auswahl von Operatoren vorgehen. Einige möchten wir im folgenden vorstellen.

**BACKUP AVOIDANCE** Menschen neigen dazu, Operatoren zu vermeiden, die den Effekt des vorangegangenen Operators aufheben [3]. Dies stellt eine recht simple Heuristik dar, die zum Ausschluss von Operatoren verwendet werden kann, jedoch nicht für die Auswahl eines Operators aus den Verbleibenden.

**HILL CLIMBING** Diese auch als Unterschiedsreduktion [2] bezeichnete Heuristik leitet Menschen oft bei der Lösung von Problemen, besonders, wenn ihnen der zugehörige Problemraum nicht bekannt ist. Hierbei wird der aktuelle Zustand mit dem Zielzustand verglichen und derjenige Operator ausgewählt, dessen Anwendung einen dem Zielzustand ähnlicheren Zustand ergibt. Dies stellt eine sehr kurzsichtige Methode dar, da immer nur die nächstmöglich erreichbaren Zustände betrachtet werden. Dieser auch als *greedy* bezeichnete Ansatz führt nicht unbedingt zur Lösung des Problems. Diese Heuristik scheitert, wenn jeder mögliche Lösungsweg einen Schritt enthält, der die Ähnlichkeit zum Zielzustand zunächst wieder reduziert.

**OPERATOR SUBGOALING** Diese Methode ist für den Fall, dass ein Operator nicht angewendet werden kann [12]. Dann besteht das aktuelle - temporäre - Ziel darin, zu ermöglichen, dass dieser Operator verwendbar wird.

**MEANS-END-ANALYSIS** Eine etwas fortgeschrittenere Methode stellt die Mittel-Ziel-Analyse dar, die unter anderem von Newell und Simon [13] untersucht wurde. Sie enthält sowohl Komponenten des *hill climbing* als auch des *operator subgoalings*. Genauso wie bei der Unterschiedsreduktion wird hier zunächst der aktuelle und der zu erreichende Zustand verglichen, um die größte Differenz zu beseitigen. Wenn jedoch ein Operator, der diese Differenz beseitigen würde, nicht unmittelbar anwendbar ist, wird er nicht verworfen, sondern die Anwendung dieses Operators wird das temporäre Ziel, das es zu erreichen gilt. Hier werden also innerhalb des Lösungsweges Teilziele formuliert – das eigentliche Problem in mehrere Teilprobleme zerlegt – um (langfristig) diejenigen Operatoren anwenden zu können, die den Unterschied zwischen aktuellem und Zielzustand reduzieren.

Auf Basis dieser bekannten Methoden haben Ragni et al. [15] ein Komplexitätsmaß für das *Rush-Hour*-Spiel vorgeschlagen, das die Anzahl der Teilziele berücksichtigt. Auf dieser Arbeit aufbauend betrachten wir mithilfe von Methoden der Netzwerkanalyse im Besonderen den Problemraum der Spiele. Die Idee, den Problemraum und die Pfade der Versuchspersonen in diesem netzwerkanalytisch zu untersuchen, ist nicht neu: Sudarshan Iyengar et al. [16] verwendeten vor allem die Idee der Zentralität, um den Problemraum des Wortspiels *word morph* zu untersuchen und fanden dabei heraus, dass Versuchspersonen sich einige Problemzustände aussuchen, über die sie dann bevorzugt navigieren.

Am Beispiel dieser Arbeiten ist das Potential von computergestützten Methoden für die kognitive Psychologie zu erkennen, das auch in dieser Arbeit genutzt wird.



# 2

---

## DAS SPIEL RUSH HOUR

---

### 2.1 DAS SPIEL

Rush Hour ist ein von Nob Yoshigahara entwickeltes und von der Firma *Thinkfun*<sup>1</sup> vertriebenes Spiel für eine Person, das auf einem quadratischen Spielbrett mit 6 mal 6 Zellen gespielt wird. Auf diesem Spielbrett werden beliebig viele Autos platziert, die genau eine Zelle breit und 2 bis 3 Zellen lang sind (Abbildung 1). Sie können horizontal in eine Reihe des Spielbrettes oder vertikal in eine Spalte des Spielbrettes platziert werden, wobei eine Zelle höchstens von einem Auto belegt werden darf. Bestimmte Konfigurationen von platzierten Autos können dann als Startkonfiguration für ein Spiel verwendet werden. Das Spielbrett besitzt einen Ausgang, der sich am rechten Rand auf Höhe der dritten Reihe von oben befindet. Ziel des Spieles ist es, zu ermöglichen, dass das Zielauto (das rote Auto in der dritten Reihe) den Ausgang des Spielbrettes erreichen kann. Es darf also keine Zelle, die sich zwischen dem Zielauto und dem Ausgang befindet, von einem anderen Auto belegt werden. Um dem Zielauto den Weg zum Ausgang freizumachen, können die Autos auf dem Spielbrett bewegt werden. Dabei kann sich jedes in seiner Reihe/Spalte beliebig vor- und zurückbewegen, aber höchstens bis es auf die Spielbrettgrenze oder ein anderes Auto trifft. Ein legaler Spielzug besteht also aus der Bewegung genau eines Autos in seiner Reihe/Spalte um beliebig viele Zellen, wobei in jedem durchlaufenen Zwischenzustand eine erlaubte Platzierung aller Autos vorliegen muss. Wir werden diese Begriffe im folgenden Abschnitt genauer einführen.

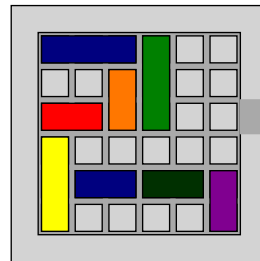


Abbildung 1: Beispiel einer Spielkonfiguration. Das rote Auto ist das Zielauto.

### 2.2 DEFINITIONEN

**SPIELBRETT** Ein *Spielbrett* definieren wir als eine Menge von Feldern:  $G = \{(i, j) \in I \times I \mid I = \{0, \dots, 5\}\}$ . Ein Feld  $g \in G$  kann entweder frei oder belegt sein.

**KONFIGURATION** Sei  $C = \{0, \dots, n\}$  die Menge der auf dem Spielbrett platzierten Autos. Wir definieren eine *Konfiguration* zu einem Spielbrett als  $k = (C, \text{pos}_k, \text{hv}_k, \text{cr}_k, \text{le}_k)$  mit  $\text{pos}_k : C \rightarrow I$ ,  $\text{hv}_k : C \rightarrow \{0, 1\}$ ,

---

<sup>1</sup> siehe auch [www.thinkfun.com](http://www.thinkfun.com)

$cr_k : C \rightarrow I$  und  $le_k : C \rightarrow \{2, 3\}$ . Auf die graphische Darstellung einer Konfiguration übertragen:  $C$  ist die Menge der auf dem Spielbrett platzierten Autos. Ein Auto kann entweder waagrecht oder senkrecht ausgerichtet sein, bei waagerechter Ausrichtung des Autos  $i \in C$  ist  $hv_k(i) = 1$ , ansonsten 0.  $le_k(i)$  gibt die Länge des Autos  $i$  an.  $cr_k(i)$  gibt für ein senkrechtes Auto die Spalte bzw. für ein waagerechtes Auto die Zeile an, in der das Auto  $i$  Felder des Spielbretts belegt.  $pos_k(i)$  gibt den minimalen Zeilenindex (für ein senkrechtes Auto) bzw. den minimalen Spaltenindex (für ein waagerechtes Auto) an, in der  $i$  Felder belegt. Betrachtet man Konfiguration  $k$  aus Abbildung 1 gilt also für das orange Auto:  $pos_k = 1$ ,  $cr_k = 2$ ,  $hv_k = 0$  und  $le_k = 2$ .

**GÜLTIGKEIT** Sei  $o : C \times G \rightarrow \{0, 1\}$  eine Abbildung, die für ein gegebenes Auto  $c \in C$  und ein gegebenes Feld des Spielbretts  $g \in G$  mit 1 angibt, dass  $g$  von  $c$  belegt wird. Wir nennen eine Konfiguration  $k$  gültig, wenn gilt:

- i)  $(\forall g \in G \forall i \in C : o(i, g) = 0) \vee$   
 $(o(j, g) = 1 \Rightarrow o(i, g) = 0 \forall i \neq j, i, j \in C).$   
 Ein Feld wird höchstens von einem Auto belegt.
- ii) Sei für  $c \in C$ :

$$G_c^* := \{(i, j) \in G \mid pos_k(c) \leq i \leq pos_k(c) + le_k(c), j = cr_k(c), \text{ falls } hv_k(c) = 1;$$

$$i = cr_k(c), pos_k(c) \leq j \leq pos_k(c) + le_k(c), \text{ falls } hv_k(c) = 0\}$$

Dann gelten  $\forall c \in C : o(c, g) = 1 \forall g \in G_c^*$  und  $o(c, h) = 0 \forall h \in G \setminus G_c^*$ .

Ein Auto belegt genau 2 oder 3 konsekutive Felder in einer Reihe oder Spalte.

- iii) es gibt ein Zielauto  $r \in C$  mit

$$hv_k(r) = 1$$

$$cr_k(r) = 2$$

$$pos_k(r) = \max\{pos_k(i), i \in C, hv_k(i) = 1 \text{ und } cr_k(i) = 2\}$$

Es gibt ein Zielauto, das waagrecht ausgerichtet ist, in der Reihe mit Index 2 steht und kein weiteres waagerechtes Auto steht rechts neben ihm.

Wenn wir im folgenden von einer Konfiguration sprechen, meinen wir implizit eine gültige Konfiguration.

**ZUG** Wir bezeichnen eine Transformation der Konfiguration  $v$  in eine neue Konfiguration  $w$  als *Zug*, wenn gilt:

- i)  $\exists j \in C : pos_v(j) \neq pos_w(j)$
- ii)  $pos_v(i) = pos_w(i) \forall i \in C, i \neq j$
- iii)  $hv_v(i) = hv_w(i) \forall i \in C$
- iv)  $cr_v(i) = cr_w(i) \forall i \in C$



v)  $le_v(i) = le_w(i) \forall i \in C$

vi) alle Konfigurationen  $z$  sind gültige Konfigurationen mit

$$z \in \{\text{pos}_v(j) < \text{pos}_z(j) < \text{pos}_w(j), j \in C, \text{pos}_v(i) = \text{pos}_z(i) \forall i \in C, i \neq j,$$

$$\forall i \in C : hv_v(i) = hv_z(i), cr_v(i) = cr_z(i), le_v(i) = le_z(i)\}$$

**BLOCKIERENDE AUTOS** Wir bezeichnen die Menge  $B_v^+(c)$  für eine Konfiguration  $v$  und ein Auto  $c \in C$  als die Menge der  $c$  blockierenden Autos in positiver Richtung:

$$B_v^+(c) = \{c' \in C \mid (hv_v(c) = hv(c') \wedge cr(c) = cr(c') \wedge \text{pos}(c') > \text{pos}(c)) \vee (hv_v(c) \neq hv_v(c') \wedge \text{pos}_v(c') \leq cr_v(c) \leq \text{pos}_v(c') + le_v(c') \wedge \text{pos}_v(c) < cr_v(c'))\}$$

Dies ist also die Menge der Autos, die verhindern, dass sich das Auto  $c$  in positiver Richtung bis an den Rand des Spielbretts bewegen kann. Analog kann  $B_v^-(c)$  als die Menge der in negativer Richtung blockierenden Autos definiert werden. Damit ist  $B_v(c) = B_v^+(c) \cup B_v^-(c)$ .

**LÖSUNGSKONFIGURATION** Eine Konfiguration  $w$  heißt *Lösungs-* oder *Endkonfiguration*, wenn  $B_w^+(\tau) = \emptyset$ . Hier fordern wir für die Eigenschaften eines Zugs zusätzlich, dass eine Lösungskonfiguration nicht durch einen Zug in eine andere Konfiguration transformiert werden kann.

**LÖSUNGSWEG** Ein *Lösungsweg* ist eine Sequenz von Zügen  $v_0 v_1 \dots v_l$ , wobei  $v_i v_{i+1}$  ein legaler Zug ist,  $v_l$  eine Lösungskonfiguration und  $v_i$  keine Lösungskonfiguration  $\forall i \in \{0, \dots, l-1\}$ .

**MENGE DER STARTKONFIGURATIONEN** Die *Menge der Startkonfigurationen*  $\mathcal{S}$  enthält alle Konfigurationen, für die eine Sequenz von Zügen existiert, die sie in eine Lösungskonfiguration transformiert.

Bevor wir mit diesen Definitionen den in der kognitiven Psychologie im Zusammenhang mit Problemlösen oft verwendeten Begriff des *Problemmaumes* formal einführen, möchten wir vorweg einige grundlegende Begriffe der Graphentheorie erläutern, die dann auf den Problemraum anwendbar sind.

**UNGERICHTETER GRAPH** Wir bezeichnen einen ungerichteten Graphen als Tupel  $G = (V, E)$  mit einer Menge  $V$  (sogenannte Knoten) und einer Menge  $E \subseteq V \times V$  (sogenannte Kanten). Hierbei ist  $E$  eine Menge ungeordneter Paare von Knoten. Wenn  $e = vw = wv \in E$  mit  $v, w \in V$ , sagen wir, dass  $v$  und  $w$  mit  $e$  inzident sind und nennen  $v$  und  $w$  Nachbarn bzw. adjazent zueinander.

**GRAD** Sei  $v \in V$ , dann ist der Grad des Knotens  $deg(v)$  definiert als die Anzahl seiner Nachbarn.

**GERICHTETER GRAPH** Ein gerichteter Graph  $G = (V, E)$ , mit Knotenmenge  $V$  besitzt eine Kantenmenge  $E \subseteq V \times V$  geordneter Paare, sodass  $(v, w) \neq (w, v)$  für  $v, w \in V$ . Für eine Kante  $e = (v, w)$  nennen wir  $v$  den Vorgänger von  $w$  und  $w$  den Nachfolger von  $v$ . Hier unterscheiden wir für einen Knoten  $v$  den *Ausgangs-* und den *Eingangsgrad*:  $\deg_{\text{out}}(v)$  bezeichnet die Anzahl der Knoten, für die  $v$  Vorgänger ist;  $\deg_{\text{in}}(v)$  die Anzahl der Knoten, für die  $v$  Nachfolger ist.

**WEG** Wir nennen die Sequenz von Knoten  $v_0 \dots v_k$  mit  $v_i \in V, i = 0, \dots, k$  und  $e_i = (v_i, v_{i+1}) \in E$  mit  $i \in \{0, \dots, k-1\}$  einen *Weg* oder *Pfad* mit Anfangsknoten  $v_0$  und Endknoten  $v_k$ . Der Weg hat die Länge  $k$ .

**PROBLEMRAUM** Wir bezeichnen den Problem- oder Zustandsraum zu einer Startkonfiguration  $s \in S$  als einen gerichteten Graphen  $G = (V, E)$ , mit  $V$  als Menge der mit legalen Zügen von  $s$  aus erreichbaren Konfigurationen, und  $E$  Menge der legalen Züge.  $V$  kann als Vereinigung der disjunkten Mengen  $\{s\}$ ,  $F$  und  $I$  gesehen werden:  $F$  enthält die durch legale Züge von  $s$  aus erreichbaren Lösungskonfigurationen,  $I$  alle intermediären Zustände. Somit ist ein Lösungsweg ein Weg durch den Problemraum mit dem Knoten  $s$  als Anfangsknoten und eine Konfiguration  $f \in F$  als Endzustand. Wir definieren die Menge  $FF \subseteq F$  als die Menge der Endkonfigurationen, deren Abstand zu  $s$  minimal ist (also die Weglänge von  $s$  zu  $f \in FF$ ). Alle Wege mit Anfangsknoten  $s$ , Endknoten  $f \in FF$  und dieser Weglänge bezeichnen wir als *optimal*. Der Außengrad eines Knoten ist in diesem Modell die Anzahl der möglichen Züge von dieser Konfiguration aus.

**SPIEL** Gegeben eine Startkonfiguration  $s \in S$ . Ein *Spiel* ist somit die Aufgabe im zugehörigen Problemraum einen Weg mit Anfangsknoten  $s$  und Endknoten  $f \in F$  zu finden.

**OPTIMALER PROBLEMRAUM** Wir bezeichnen den Subgraphen  $G_o := (V_o, E_o)$  von  $G$  als *optimalen Problemraum*, wenn  $V_o$  alle Knoten und  $E_o$  alle Kanten enthält, die im Problemraum in einem optimalen Weg enthalten sind.

**KOMPLEXITÄTSMASß** Wir definieren ein Komplexitätsmaß als eine Abbildung  $\mathcal{C} : S \rightarrow \mathbb{R}$ , die eine Startkonfiguration auf eine reelle Zahl abbildet und die Komplexität, einen Lösungsweg zu dieser Startkonfiguration zu finden, nach einem gegebenen Kriterium darstellt.

Nachdem wir die nötigen Begriffe geklärt haben, widmen wir uns der theoretischen Untersuchung der Komplexität der vom Hersteller herausgegebenen Spielen.

# 3

---

## UNTERSUCHUNG DER LEVELKARTEN

---

### 3.1 DATEN

Um die Komplexität von Startkonfigurationen zu untersuchen, verwenden wir als Menge der Startkonfigurationen die Kartensätze des Herstellers. Dieser hat drei verschiedene Editionen sowie drei Zusatzkartensätze herausgebracht; wir werden sowohl die Levelkarten der Standardedition, der *Deluxe*-Edition sowie der *Junior*-Edition als auch die drei Zusatzkartensätze untersuchen. Jeder Kartensatz enthält 40 (*deluxe*: 60) verschiedene Startkonfigurationen, denen jeweils ein Schwierigkeitsgrad zugeordnet ist. Die Schwierigkeitsabstufungen des Herstellers lauten *beginner* (B), *intermediate* (I), *advanced* (A), *expert* (E) und *grand master* (G). Diese vorgegebenen Schwierigkeitsgrade verwenden wir als Vergleichskriterium, um unsere Komplexitätsmaße zu evaluieren.

Einige Startkonfigurationen mussten entfernt werden, da sie zwei Zielautos enthielten mit der Aufgabe, beide vom Spielbrett zu entfernen. Um bei einem einheitlichen und besser vergleichbaren Ziel zu bleiben, wurden diese Startkonfigurationen im Folgenden nicht mehr berücksichtigt. Zudem finden sich identische Konfigurationen in den Kartensätzen, die ebenfalls entfernt wurden. Tabelle 1 zeigt die Verteilung der Startkonfigurationen auf die Schwierigkeitsgrade.

	B	I	A	E	G	SUM
Junior	10	10	10	10	0	40
Standard	10	10	10	10	0	40
Deluxe	10	3	3	3	8	27
Kartensatz 2	0	10	9	6	7	32
Kartensatz 3	0	10	10	9	9	38
Kartensatz 4	0	10	10	8	8	36
SUM	20	43	42	36	32	173

Tabelle 1: Anzahl an verwendeten Startkonfigurationen pro Kartensatz von *Thinkfun*, geordnet nach dessen Schwierigkeitskategorien: beginner (B), intermediate (I), advanced (A), expert (E), grand master (G).

Wir vermuten, dass die *Junior*-Edition gesondert betrachtet werden sollte, da sie für Kinder konzipiert wurde und die dort vom Hersteller angegebenen Schwierigkeitsangaben nicht mit den der übrigen Editionen übereinstimmen. Da-

her werden wir die Maße für diese Konfigurationen zwar berechnen, sie aber bei der weiteren Analyse nicht berücksichtigen. Die vorliegenden Ergebnisse können dann in weiterführenden Untersuchungen betrachtet werden.

### 3.2 KOMPLEXITÄTSMASSE

Wir stellen hier die Komplexitätsmaße vor, die wir in den folgenden Kapiteln untersuchen werden. Die Hypothese ist, dass die entwickelten Komplexitätsmaße mit der von *Thinkfun* vorgeschlagenen Komplexität korrelieren. Einige der vorgestellten Maße basieren auf der Untersuchung des Zustands- oder des optimalen Graphen der Startkonfiguration, ohne die jeweiligen Spielkonfigurationen zu berücksichtigen, sodass nicht relevant ist, für welches Spiel (ob Rush Hour oder ein anderes Brettspiel) das Maß berechnet wird. Hier wäre zum Beispiel das Maß *Größe des Problemraums* anzuführen. Andere Maße stützen sich auf die spezifischen Eigenschaften des *Rush-Hour*-Spiels, wie beispielsweise das Maß *Anzahl der Autos auf dem Spielbrett*. Eine Übersicht über die folgenden Maßen kann Tabelle 3 entnommen werden.

**GRÖÖE DES ZUSTANDSRAUMES**    Dieses Maß basiert auf der Überlegung, dass die Lösung einer Startkonfiguration als schwieriger empfunden werden sollte, je größer die Anzahl der erreichbaren Zustände ist. Hierfür definieren wir drei Maße: die Anzahl von Knoten im Zustandsraum (der Startzustand, alle erreichbaren Endzustände und alle erreichbaren Zwischenzustände) (*nodes*), die Anzahl der Knoten im Zustandsraum ohne die Endzustände (*nodf*), und die Anzahl der Kanten im Zustandsraum, also die Anzahl möglicher Züge, (*edges*). Hierbei werden die reziproken gerichteten Kanten  $e = (v, w)$  und  $g = (w, v)$  zwischen zwei Knoten  $v, w \in V$  als eine Kante gezählt.

**LÄNGE DES KÜRZESTEN WEGES**    Eine Startkonfiguration könnte als schwieriger empfunden werden, je mehr Züge nötig sind, um das Spiel zu lösen. Entsprechend dieser Intuition definieren wir die beiden Komplexitätsmaße *length of shortest path* (*lsp*) und *average length of shortest path* (*avlsp*). *lsp* ist die minimale Anzahl an Zügen, die benötigt wird, um die Startkonfiguration in einen Endzustand zu transformieren. Der Zustandsraum enthält sowohl Endzustände, die innerhalb der minimalen Anzahl an Zügen erreichbar sind, als auch Endzustände, die, auch wenn der kürzeste Weg von dem Startzustand zu diesem Endzustand gewählt wird, nicht innerhalb dieser Anzahl an Zügen erreichbar sind.

Diese Tatsache soll das Maß *avlsp* widerspiegeln, das folgendermaßen definiert ist: Sei  $lsp(f)$  mit  $f \in F$ ,  $F$  Menge der Endzustände, die Länge des kürzesten Weges vom Startzustand  $s$  zum Endzustand  $f$ . Dann ist

$$avlsp = \frac{1}{|F|} \sum_{f \in F} lsp(f),$$

also die durchschnittliche Anzahl an Zügen zu einem Endzustand.

ANZAHL AN ENTSCHEIDUNGSMÖGLICHKEITEN Wir vermuten, dass eine Startkonfiguration für eine Versuchsperson schwieriger zu lösen ist, je mehr Möglichkeiten in jedem Zug zur Auswahl stehen. Dazu führen wir zunächst das Maß *average degree (avdg)* ein, das den durchschnittlichen Grad der Knoten im Zustandsraum darstellt. Hierbei werden nur ausgehende Kanten und die Knoten  $V \setminus F$  berücksichtigt, da in den Endzuständen keine Wahl mehr getroffen werden muss. Wir definieren also

$$\text{avdg} = \frac{1}{|V \setminus F|} \sum_{v \in V \setminus F} \text{deg}_{\text{out}}(v)$$

Da wir jedoch annehmen, dass die meisten Knoten des Zustandsraumes nur selten von Spielern in ihrem Lösungsweg verwendet werden, sondern nur ein sehr kleiner Teil der Zustände tatsächlich besichtigt wird und diese sich im Idealfall entlang des optimalen Pfades häufen, möchten wir diesen Überlegung ebenfalls in einem Komplexitätsmaß abbilden. Dazu definieren wir das Maß *average degree on optimal paths (avdgop)* als durchschnittlicher ausgehender Grad der Knoten im optimalen Graphen, also

$$\text{avdgop} = \frac{1}{|V_o \setminus F_o|} \sum_{v \in V_o \setminus F_o} \text{deg}_{\text{out}}(v)$$

Hier werden allerdings nur die ausgehenden Kanten eines Knoten betrachtet, die im optimalen Graphen enthalten sind. Im Maß *average degree on optimal paths in graph (avdgog)* ziehen wir deshalb für alle Knoten auf optimalen Wegen die Gesamtanzahl ausgehender Kanten im Problemraum in Betracht:

$$\text{avdgog} = \frac{1}{|V_o \setminus F_o|} \sum_{v \in V_o \setminus F_o} |\{(v, w) \in E\}|$$

Wir beobachten jedoch, dass von vielen Zuständen mehrere Kanten ausgehen, die zu einem optimalen Pfad gehören. Intuitiv ist klar, dass die Entscheidung, welcher Zug ausgeführt wird, nicht schwierig ist, wenn zwar viele Züge möglich sind, viele jedoch zu einem optimalen Weg gehören. Auch scheint es offensichtlich, dass die Entscheidung als besonders schwierig empfunden werden kann, wenn von vielen möglichen Zügen nur einer „richtig“ ist. Aus diesen Überlegungen heraus führen wir die *branching complexity (br)* ein: der durchschnittliche Anteil an ausgehenden Kanten, die nicht zu optimalen Wegen gehören. Wir definieren also die *branching complexity* für einen Knoten  $v \in V$  als

$$\text{br}(v) = 1 - \frac{|\{(v, w) \in E_o \mid w \in V_o\}|}{|\{(v, u) \in E \mid u \in V\}|}$$

und können damit die *branching complexity* für ein Spiel als den Durchschnitt über alle Knoten des optimalen Graphs definieren:

$$\text{br} = \frac{1}{|V_o \setminus F_o|} \sum_{v \in V_o} \text{br}(v)$$

Es ist offensichtlich, dass  $\text{br} : S \rightarrow [0, 1]$ . Die zugrunde liegende Hypothese ist, dass, je größer  $\text{br}$  ist, desto schwieriger sollte das Spiel empfunden werden. Für

alle Knoten  $t \in V \setminus V_o$  gilt  $br(t) = 1$ . Da wir auch hier wieder vermuten, dass die meisten Spieler nur durch einen Teil des Zustandsraumes navigieren werden, dessen Wege nicht zu weit von denen des optimalen Graphen abweichen, wird die vorherige Durchschnittsbildung nur über den optimalen Graphen und nicht über den gesamten Zustandsraum ausgeführt.

**ANZAHL OPTIMALER WEGE** Um nicht nur die Länge der optimalen Wege, sondern auch ihre Anzahl zu berücksichtigen, führen wir das Maß *shortest paths* ( $sp$ ) ein, das definiert ist als die Anzahl kürzester Wege vom Startzustand  $s$  zu einem Endzustand  $f \in F_o$ . Zusätzlich definieren wir das Maß *shortest paths per final state* ( $sppf$ ) wie folgt:

$$sppf = \frac{sp}{|F_o|}$$

**SPIELEIGENSCHAFTEN** Die bisherigen Maße basierten auf dem Problemraum des Spiels, sodass sie auch auf andere Brettspiele angewendet werden könnten. Es ist jedoch wünschenswert, dass nicht nur die Eigenschaften der Spielnetzwerke berücksichtigt werden, sondern auch besondere Eigenschaften der betrachteten Startkonfiguration.

Ein grundlegender Ansatz betrachtet die Anzahl der Autos auf dem Spielbrett, denn je mehr Autos auf dem Spielbrett platziert sind, desto mehr Objekte müssen bei der Lösung der Konfiguration beachtet werden und desto mehr Abhängigkeiten sind potentiell aufzulösen. Jedoch ist auch ersichtlich, dass bei einer größeren Anzahl an Autos weniger Spielzüge möglich sind, was es wieder einfacher machen könnte. Daher führen wir zunächst die beiden für ein Spiel statischen Maße *cars* und *fields* ein, die die Anzahl der auf dem Brett platzierten Autos und, um die Länge der Autos zu berücksichtigen, die Anzahl der belegten Felder misst. Um der Vermutung nachzugehen, dass bei einer größeren Anzahl an Autos die Wahlmöglichkeiten in jedem Zug weniger werden, definieren wir die beiden Maße *moving cars* ( $mc$ ) und *moving cars in optimal graph* ( $mcop$ ): Für eine Konfiguration  $v$  sei  $mc(v)$  die Anzahl der Autos, die sich in dieser Konfiguration um mindestens ein Feld bewegen können.  $mc$  für ein Spiel ist dann der Mittelwert  $mc = \frac{1}{|V \setminus F|} \sum_{v \in V \setminus F} mc(v)$ . Analog zur Begründung oben bilden wir auch den Durchschnitt über alle Knoten des optimalen Graphen und erhalten

$$mcop = \frac{1}{|V_o \setminus F_o|} \sum_{v \in V_o \setminus F_o} mc(v).$$

Diese Maße werden den Maßen *avdg* und *avdgop* ähneln, jedoch berücksichtigen sie, dass im ausgehenden Grad oft mehrere Kanten vorkommen, die die Bewegung desselben Autos betreffen. Da es aber oft für den weiteren Spielverlauf keine Rolle spielt, um wieviele Felder ein Auto nun bewegbar ist, betrachten wir in diesem Maß nur die Anzahl der überhaupt bewegbaren Autos.

**UNINTUITIVE ZÜGE** In Kapitel 1 haben wir sowohl die Mittel-Ziel-Analyse als auch die Unterschiedsreduktion beschrieben. Wir möchten diese im folgenden Maß berücksichtigen. Wir bezeichnen einen Spielzug als *unintuitiv*, wenn in ihm (i) das Zielauto rückwärts bewegt wird oder (ii) ein zusätzliches Auto vor das

Zielauto  $r$  geschoben wird und es somit blockiert. Formaler können wir wie folgt definieren: Seien  $v, w \in V$  Spielzustände und  $e = (v, w) \in E$ , dann bezeichnen wir den Zug  $e$  als *counterintuitive move* ( $cm$ ), wenn  $\text{pos}_v(r) > \text{pos}_w(r)$  oder  $|B_v^+(r)| < |B_w^+(r)|$  gilt, und definieren das Komplexitätsmaß  $cm$  als die durchschnittliche Anzahl an unintuitiven Zügen, die während eines optimalen Lösungsweges gemacht werden müssen. Da wir jedoch beobachten, dass es Kanten gibt, die in vielen optimalen Lösungswegen enthalten sind, sollten diese auch stärker gewichtet werden, wenn es sich um einen unintuitiven Zug handelt. Daher definieren wir unser Komplexitätsmaß  $cm$  wie folgt: Sei  $\omega : E \rightarrow \mathbb{N}$ , so dass  $\omega(e)$  angibt, in wievielen optimalen Wegen die Kante  $e$  vorkommt, und  $E_c = \{e \in E_o \mid e \text{ ist unintuitiv}\}$ . Dann ist

$$cm = \frac{\sum_{e \in E_c} \omega(e)}{sp}.$$

Da zu erwarten ist, dass ein längerer Lösungsweg tendenziell mehr unintuitive Züge enthält, und wir diesen potentiellen Komplexitätsfaktor bereits in unserem Maß  $lsp$  abgebildet haben, führen wir zusätzlich das Maß *counterintuitive moves per path length* ( $cmpl$ ), das wir als  $cmpl = \frac{cm}{lsp}$  definieren.

### 3.3 IMPLEMENTIERUNG

Wir fassen in diesem Kapitel kurz zusammen, wie wir die Berechnung der oben definierten Komplexitätsmaße anhand unserer Startkonfigurationen implementiert haben. Einzige Eingabe des Programms sind die 173 Startkonfigurationen, die uns in handelsüblicher Papierform vorlagen. Ausgabe ist eine Datei, die die 17 berechneten Maße jeweils für alle Startkonfigurationen enthält, mit der dann weitere Analysen erfolgen können.

Die Funktionalitäten der erforderlichen Implementierung können wie folgt gegliedert werden:

- i) Digitalisierung der Levelkarten
- ii) geeignete Darstellung der Konfigurationen
- iii) Berechnung des Problemraums sowie des optimalen Problemraums
- iv) Berechnung der Komplexitätsmaße auf Basis der Problemräume sowie der Spiellogik

**KODIERUNG DER SPIELKONFIGURATIONEN** Es gibt einige Möglichkeiten, eine Spielkonfiguration eindeutig und effizient darzustellen, jedoch haben wir uns für eine Kodierung entschieden, die der Definition einer Konfiguration aus Kapitel 2 ähnlich ist.

Wir speichern Informationen über eine Konfiguration, die sich durch legale Züge nicht ändern können, global, sodass in der Kodierung einer Konfiguration nur Informationen enthalten sind, die nicht konstant sind.

Zu den globalen Eigenschaften gehören offensichtlich die Anzahl, Länge und Ausrichtung der Autos, sowie Reihe (bei horizontalen Autos) bzw. Spalte (bei vertikalen Autos). Die Reihen und Spalten einer Konfiguration erhalten eine fortlaufende Nummerierung von 0 bis 5 (von links nach rechts, von oben nach unten). Ebenso werden die auf einem Spielbrett vorhandenen Autos, beginnend bei

MAß	DEFINITION/ERKLÄRUNG	WERTE
<i>nodes</i>	Anzahl der Knoten im Problemraum	$\mathbb{N}$
<i>nodf</i>	Anzahl der Knoten ohne Endzustände	$\mathbb{N}$
<i>edges</i>	Anzahl der Kanten im Problemraum	$\mathbb{N}$
<i>lsp</i>	Länge des optimalen Lösungsweges	$\mathbb{N}$
<i>avls</i>	durchschnittliche Länge der Lösungswege	$\mathbb{R}_{\geq 0}$
<i>avdg</i>	Durchschnittsgrad der Knoten im Problemraum	$\mathbb{R}_{\geq 0}$
<i>avdgop</i>	Durchschnittsgrad der Knoten im optimalen Graphen	$\mathbb{R}_{\geq 0}$
<i>avdgog</i>	Durchschnittsgrad der Knoten auf optimalen Pfaden im Problemraum	$\mathbb{R}_{\geq 0}$
<i>br</i>	durchschnittlicher Anteil der Kanten, die vom optimalen Graphen wegführen	$[0, 1]$
<i>sp</i>	Anzahl kürzester Wege von $s$ zu $f \in F_o$	$\mathbb{N}$
<i>sppf</i>	Anzahl kürzester Wege pro Endzustand	$\mathbb{R}_{\geq 0}$
<i>cars</i>	Anzahl der Autos	$\{0, \dots, 18\}$
<i>fields</i>	Anzahl belegter Felder	$\{0, \dots, 36\}$
<i>mc</i>	durchschnittliche Anzahl bewegbarer Autos	$\mathbb{R}_{\geq 0}$
<i>mcop</i>	durchschnittliche Anzahl bewegbarer Autos im optimalen Graphen	$\mathbb{R}_{\geq 0}$
<i>cm</i>	gewichtete Anzahl unintuitiver Züge	$\mathbb{R}_{\geq 0}$
<i>cmpl</i>	gewichtete Anzahl unintuitiver Züge im Verhältnis zur Lösungslänge	$\mathbb{R}_{\geq 0}$

Tabelle 2: Eine Zusammenfassung der im weiteren Verlauf der Arbeit verwendeten Komplexitätsmaße



0, durchnummeriert. Das Auto, das zum Ausgang muss, erhält die Nummer 0, danach werden alle horizontalen Autos durchnummeriert (von oben nach unten, von links nach rechts), danach alle vertikalen Autos (von links nach rechts, von oben nach unten). Global werden dann drei Integer-Arrays gespeichert, die jeweils zu jedem Auto diese Information enthalten (sei  $n$  die Anzahl der Autos in den Konfigurationen):

- *HV* ist ein Integer-Array der Länge  $n$ , das am  $i$ . Index bzw. der  $i$ . Position die Information enthält, wie das  $i$ . Auto ausgerichtet ist. *HV* enthält eine 1, wenn das entsprechende Auto horizontal liegt, oder eine 0, falls die Ausrichtung vertikal ist.
- *CR* repräsentiert eine Zuordnung der Autos auf ihre unveränderliche Spalte bzw. Reihe. Dazu enthält *CR* in der gleichen Reihenfolge wie die Autos durchnummeriert wurden, die Nummer der Reihe des obersten Feldes (bei vertikalen Autos) bzw. die Nummer der Spalte des am weitesten links liegenden Feldes (bei horizontalen Autos), die durch das Auto belegt wird.
- *length* ist ebenfalls ein Array der Länge  $n$ , das in der Reihenfolge wie die Nummerierung der Autos die Länge des jeweiligen Autos enthält, also die Anzahl der Zellen, die das Auto belegt.

Eine einzelne Konfiguration in diesem Kontext wird nun ebenfalls als ein Array der Länge  $n$  gespeichert: In der Reihenfolge, in der die Autos durchnummeriert wurden, kodiert jeder Eintrag die Position des jeweiligen Autos: bei horizontalen Autos die Spaltennummer des am weitesten links liegenden Feldes, das vom Auto belegt wird, und bei vertikalen Autos die Reihenummer des obersten Feldes, das vom Auto belegt wird.

Die Kodierung der Konfiguration in **1** lautete beispielsweise 0013 3104 und die folgenden Arrays würden gespeichert:

```

HV      1111 0000
CR      2044 0235
length  2322 3232

```

Diese Kodierung bietet sich vor allem an, wenn innerhalb eines Spiels Berechnungen durchgeführt werden, sodass bei Bedarf auf die globalen Eigenschaften zugegriffen werden kann, diese jedoch nicht in jedem Knoten mitgespeichert werden müssen. Zudem ermöglicht diese Kodierung prinzipiell auch einem Menschen (mit Übung) die aktuelle Autokonfiguration an der Kodierung abzulesen, wenn die globalen Eigenschaften bekannt sind. Offensichtlich ist auch, dass in einem legalen Zug sich genau eine Ziffer der Kodierung ändert und somit abgelesen werden kann, welches Auto sich in einem Zug um wieviele Felder bewegt hat.

Da mit der Positionskodierung für jede einzelne Konfiguration ein Integerarray der Dimension  $n$  ( $n$  Anzahl der Autos) gespeichert wird und damit in einigen Fällen viel Speicherplatz benötigt wird, wird zur Speicherung der Spielkonfigurationen eine andere Kodierung verwendet, die es erlaubt, die Zustände speicherplatzeffizienter zu verwalten. Diese im Programmcode als Long-ID bezeichnete Kodierung basiert auf folgender Idee:

Die globalen Eigenschaften *number cars*, *HV*, *CR* und *length* werden beibehalten wie bei der vorherigen Darstellung. Die veränderlichen Informationen eines

Zustandes werden in 60 Bit gespeichert (sodass ein long zur Speicherung eines Zustands genügt):

Die ersten 6 Bit enthalten binär die Anzahl der vorhandenen Autos (bei einem Feld von  $6 * 6 = 36$  Feldern und Autos mit minimaler Länge 2 können maximal 18 Autos auf einem Brett stehen, es werden also maximal 6 Bit benötigt). Die restlichen 54 Bit werden genutzt, um in jeweils 3 Bit die Positionen der Autos (analog zur Positions-ID) zu speichern. Die letzten 3 Bit enthalten die Position des Autos 0, die vorletzten 3 Bit die des Autos 1, etc. Bei weniger als 18 Autos werden die übrigen bits bei 0 belassen.

Diese Kodierung ermöglicht es, die Problemgraphen speicherplatzeffizient zu verwalten sowie die Kodierung einer Konfiguration schnell in eine menschenlesbare Form zu bringen.

**DIGITALISIERUNG DER STARTKONFIGURATIONEN** Da es die zeitlichen Grenzen dieser Arbeit weit überschreiten würde, einen Ansatz aus der Bildverarbeitung zu implementieren, haben wir einen zeiteffizienten Weg gefunden, die Position der Autos händisch einzugeben und diese dann in die obige Kodierung umrechnen zu lassen. Um eventuelle Eingabefehler zu minimieren, wurde eine Klasse implementiert, die die Informationen über eine Konfiguration wieder graphisch darstellen kann, damit eine schnelle Fehlererkennung stattfinden konnte.

**BERECHNUNG DES PROBLEMRAUMS** Im Prinzip kann der Problemraum durch eine einfache Breitensuche beginnend bei der Startkonfiguration exploriert werden. Um die möglichen Nachfolgezustände einer Konfiguration zu berechnen, wird in einer gesonderten Klasse eine simple Spiellogik implementiert, die zum Beispiel für einen gegebenen Zustand und den Index eines Autos prüft, ob es sich auf dem Spielbrett bewegen kann, oder einen Zustand daraufhin überprüft, ob es sich um einen Lösungszustand handelt. Wird ein Lösungszustand erreicht, werden für diesen keine weiteren Nachfolger berechnet. So wird sichergestellt, dass der errechnete Problemraum mit unserer Definition konsistent ist, da von einem Endzustand keine weiteren Züge nötig sind. Ansonsten könnten theoretisch noch weitere neue Konfigurationen entdeckt werden. Da wir den Problemraum als gerichteten Graphen verstehen, heißt das insbesondere, dass es von Endzuständen – im Gegensatz zu allen anderen Zuständen – keine ausgehenden Kanten gibt.

Für die Speicherung des Graphs verwenden wir eine Adjazenzliste in Form einer HashMap, die als Schlüssel einen Zustand als Long enthält und als Wert ein HashSet mit den Nachfolgern des Zustands, ebenfalls jeweils als Long gespeichert. Dies ermöglicht uns schnellen Zugriff auf die Knoten und ihre Nachbarn sowie die effiziente Überprüfung auf Nachbarschaft von Knoten.

Um den optimalen Problemraum zu erstellen, verwenden wir einen erweiterten Dijkstra-Algorithmus [4], der bei der Startkonfiguration beginnend die Entfernung von allen Knoten des Problemraums sowie ihre Vorgänger auf dem kürzesten gefundenen Weg speichert. Jedoch erlauben wir, dass ein Knoten mehrere Vorgänger haben darf, falls diese Vorgänger alle dieselbe (minimale) Entfernung zum Startknoten besitzen, um darzustellen, dass ein Knoten auf mehreren optimalen Wegen durch den Problemraum erreicht werden kann. Um aus diesen Informationen den optimalen Problemraum zu erstellen, beginnen wir bei allen Endzuständen, die

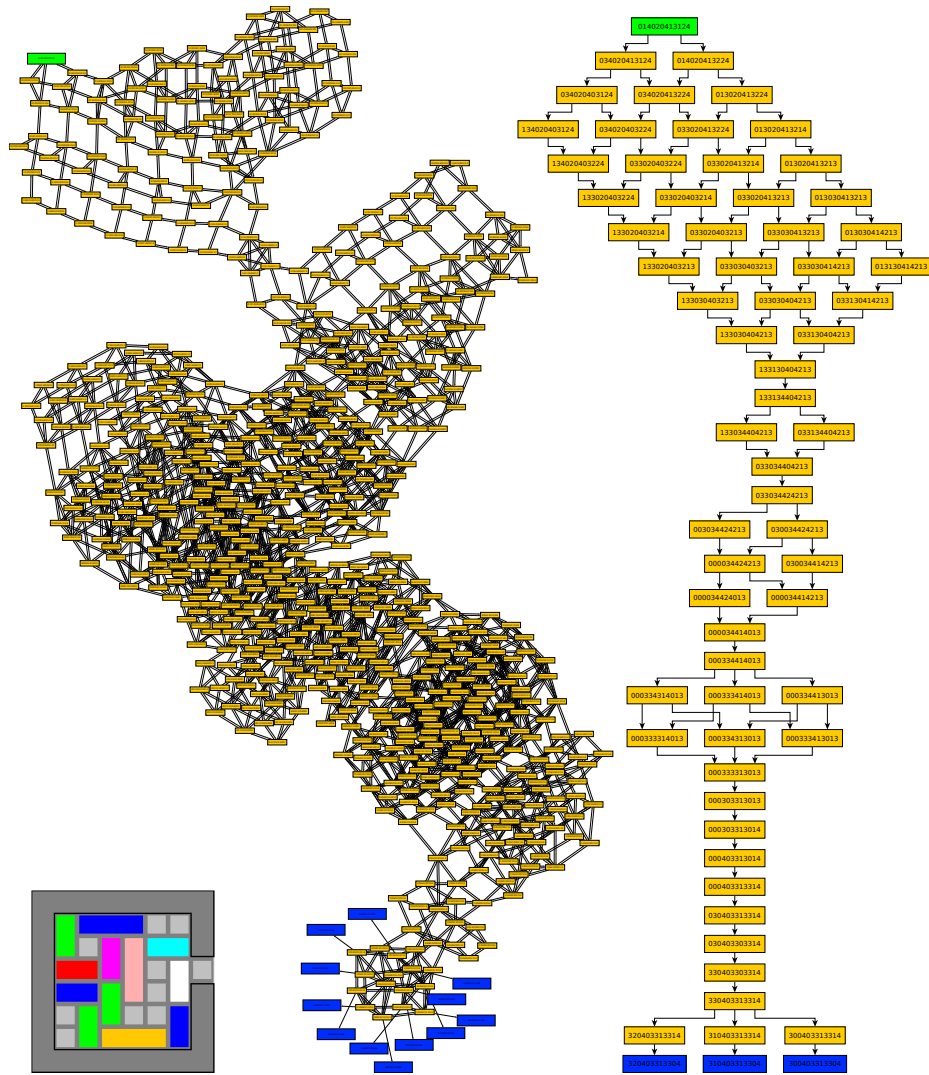


Abbildung 2: Beispiel für einen Problemraum (Mitte) und den dazugehörigen optimalen Problemraum (rechts) zu einer Startkonfiguration (links). Der Startzustand ist grün, erreichbare Endzustände sind blau gefärbt.

in optimaler Zuganzahl erreichbar sind und bewegen uns „rückwärts“ durch den Graphen und fügen alle Knoten und Kanten, die bei der Exploration des Graphen gesichtet werden, dem optimalen Graphen hinzu, den wir in der gleichen Struktur wie den Problemraum speichern. Innerhalb von  $k$  Iterationen ( $k$  Länge des kürzesten Weges vom Startzustand zu einem Endzustand  $f \in FF$ ) ist der optimale Graph erstellt.

Um eventuelle weitere Untersuchungen auf diesen Graphen zu ermöglichen, erzeugen wir sowohl Kantenlisten als auch GML-Dateien der beiden Graphen. Ein Beispiel für den Problem- und den optimalen Problemraum zu einer gegebenen Startkonfiguration findet sich in Abbildung 2. Die Graphen aller Startkonfigurationen finden sich auf der beigelegten CD.

BERECHNUNG DER KOMPLEXITÄTSMASSE Da sowohl die Problemräume als auch die optimalen Graphen sowie die Spielloge bereits implementiert sind, beinhaltet die Berechnung der beschriebenen Komplexitätsmaße keinerlei Neuheiten.

### 3.4 ERGEBNISSE

Wir erhalten für alle Startkonfigurationen eine Liste der definierten Maße, die wir für die folgende Analyse verwenden können. Wir führen einige einfache Analysemethoden auf den vorliegenden Daten aus, um einen Vergleich mit der Schwierigkeitseinteilung des Herstellers zu ermöglichen.

#### 3.4.1 Verteilungsanalyse

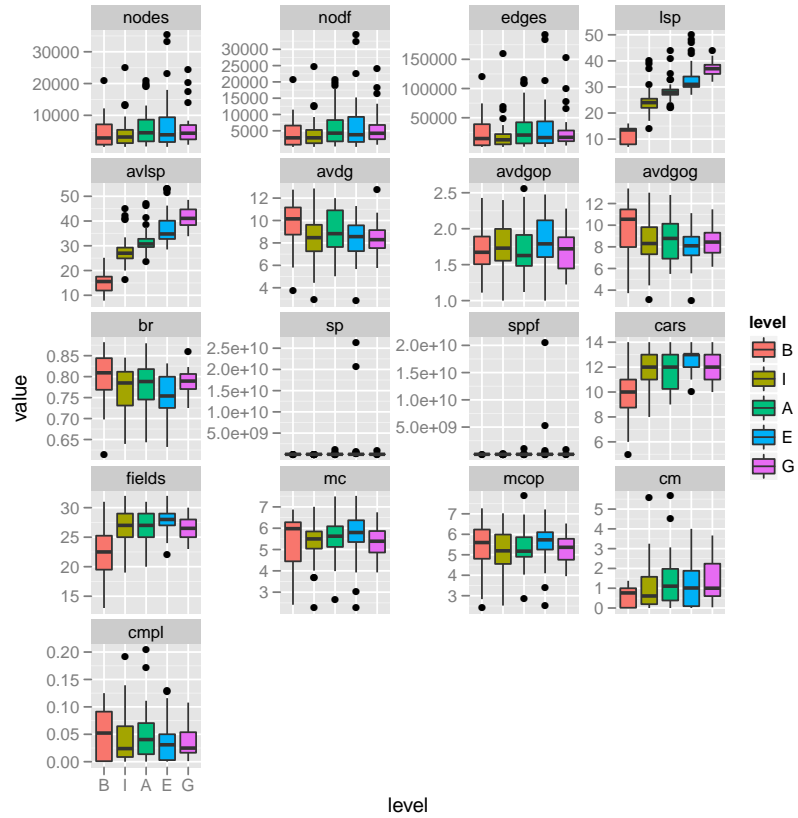
GRUPPIERUNG NACH SCHWIERIGKEITSANGABE Wir betrachten zunächst die Verteilung und den Durchschnitt der Maße gruppiert nach Schwierigkeitseinteilung des Herstellers. Für jedes unserer Maße tragen wir mithilfe eines Boxplots seine Werte auf, gruppiert nach Levelangabe des Herstellers (*beginner*, *intermediate*, *advanced*, *expert*, *grand master*) auf. Innerhalb eines Levels wird über alle Spiele gemittelt: jede Box enthält die mittleren 50 % der Daten, der Querbalken stellt den Median der Daten dar. Die Höhe der Box entspricht dem Interquartilsabstand und ist damit eine Visualisierung der Streuung der Daten. Die Whiskers (Schnurrhaare) zeigen das Minimum bzw. das Maximum der Datenpunkte an, solange sie innerhalb des 1,5-fachen Interquartilsabstand liegen. Andernfalls werden sie als einzelne Datenpunkte dargestellt und zeigen somit Ausreißer an [6].

Würde ein Maß perfekt mit der Schwierigkeitsangabe des Herstellers korrelieren, sähen wir sehr flache Boxen ohne Whiskers, die eine aufsteigende Gerade bildeten und deren Wertebereich sich nicht überschneite. Betrachten wir Abbildung 3a: Zunächst fällt auf, dass diese perfekte Korrelation bei kaum einem der Maße zu sehen ist. Nur in den Maßen *lsp* und *avlsp*, die beide die Länge der Lösungswege widerspiegeln, ist eine Korrelation zu erkennen. Die Länge des optimalen Weges und die durchschnittliche Länge der kürzesten Wege zu einer Endkonfiguration steigen direkt mit der Schwierigkeitseinschätzung des Herstellers. Dies stimmt auch mit der Beobachtung Ragnis et. al. [15] überein. Allerdings fällt auf, dass vor allem in den mittleren Gruppen *intermediate*, *advanced* und *expert* der Interquartilsabstand zwar klein ist, es jedoch eine beträchtliche Anzahl an Ausreißern gibt, sodass sogar in den *intermediate*-Leveln Spiele existieren, deren minimale Lösung länger ist als die durchschnittliche minimale Lösungslänge von *grand master*-Spielen. Dies ist erstaunlich und sollte genauer untersucht werden, da es dann entweder noch andere Faktoren geben muss, die diese Spiele wieder einfacher oder komplexer machen oder die Einteilung durch den Hersteller in die fünf Schwierigkeitsgruppen nicht perfekt ist.

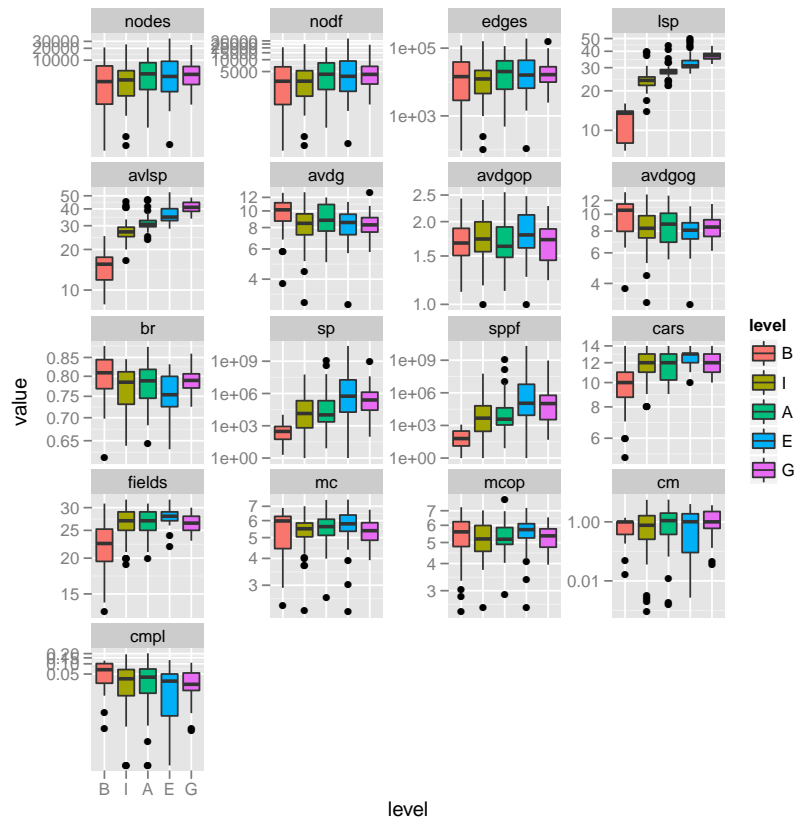
Weiterhin ist zu erkennen, dass die Größe des Zustandsraumes (*nodes*) keinen Einfluss auf die Schwierigkeit zu haben scheint (oder zumindest vom Hersteller nicht bei der Einteilung berücksichtigt wurde), der Median aller Level ist zwar fast ausschließlich aufsteigend, der Mittelwert ausschließlich aufsteigend (vergleiche auch Tabelle 3), jedoch kann bei einer solchen Streuung der Daten, sichtbar an der

		BEGINNER	INTERMEDIATE	ADVANCED	EXPERT	GRAND MASTER
<i>nodes</i>	Mean	3613.433	3800.17	5561.962	6212.891	6379.958
	SD	4645.748	4364.602	5932.016	7779.522	6292.91
<i>nodf</i>	Mean	3430.033	3646.132	5367.615	5983.022	6125
	SD	4481.245	4257.328	5735.614	7508.849	5956.503
<i>edges</i>	Mean	19419.23	18332.58	27910.27	29349.67	29748.88
	SD	26817.77	25978.05	32968.3	41649.18	35887.87
<i>lsp</i>	Mean	10.1	21.49057	25.76923	31	37.25
	SD	3.707867	7.586989	6.676313	7.260242	3.179418
<i>avlsp</i>	Mean	13.02751	24.91696	29.2251	34.27792	41.05664
	SD	4.868809	8.204732	7.148574	8.300011	3.888242
<i>avdg</i>	Mean	9.316227	8.345528	8.878442	8.267407	8.441517
	SD	2.39478	1.988619	1.960545	1.854871	1.540604
<i>avdgop</i>	Mean	1.59976	1.720421	1.677477	1.749565	1.70635
	SD	0.3558663	0.3513089	0.331901	0.3374794	0.2974596
<i>avdgog</i>	Mean	9.446583	8.413936	8.551575	8.129972	8.461379
	SD	2.591612	1.888797	1.974222	1.763019	1.418847
<i>br</i>	Mean	0.8078133	0.7789151	0.7811327	0.7659848	0.7860958
	SD	0.05805131	0.05357323	0.05325509	0.05312915	0.03240045
<i>sp</i>	Mean	989.1333	2708525	30051880	1055018000	40683410
	SD	2493.625	11028490	159122200	4878146000	187725400
<i>sppf</i>	Mean	174.0306	1350731	23772900	5.843728	40272220
	SD	309.2765	7978521	151017900	3113466000	187814300
<i>cars</i>	Mean	8.6	11.03774	11.30769	11.97826	11.875
	SD	2.594424	2.369374	1.809977	1.79492	1.075922
<i>fields</i>	Mean	19.93333	25.22642	25.71154	26.95652	26.625
	SD	5.24525	4.697141	3.477302	3.37281	1.789158
<i>mc</i>	Mean	4.893373	5.152194	5.418979	5.46303	5.373513
	SD	1.413812	1.111941	0.9919055	1.217084	0.720149
<i>mcop</i>	Mean	4.765443	5.029881	5.115194	5.289702	5.283546
	SD	1.480722	1.152404	1.007147	1.105065	0.747815
<i>cm</i>	Mean	0.6410233	0.7514321	0.8692808	0.9488891	0.9756542
	SD	0.6076613	0.8185717	1.020323	1.173705	0.7905551
<i>cmpl</i>	Mean	0.06521	0.03941698	0.03465385	0.03253478	0.026575
	SD	0.07886911	0.04789213	0.0402432	0.03967863	0.02225896

Tabelle 3: Mittelwert und Standardabweichung der vorgestellten Maße, gruppiert nach Schwierigkeitseinteilung des Herstellers.



(a) Boxplots, gruppiert nach Schwierigkeitsangabe



(b) Boxplots, gruppiert nach Schwierigkeitsangabe, mit logarithmischer Skala

Abbildung 3: Boxplots aller Startkonfigurationen (außer der Junioredition und Dopplungen)

großen Standardabweichung, zwar von einer Tendenz, aber von keiner Erkenntnis gesprochen werden.

Bei den auf dem Grad der Knoten basierenden Maßen ist weder eine positive noch eine negative Korrelation zu erkennen. Bei den Maßen *avdg*, *avdgog* und *br* könnte eine negative Korrelation vermutet werden, diese kann jedoch aufgrund dieser Ergebnisse weder bestätigt noch widerlegt werden. Allerdings ist auffällig, dass die *beginner*-Level sich stärker von den restlichen Leveln absetzen und, obwohl sie weniger Datenpunkte enthalten, eine größere Streuung aufweisen.

Betrachten wir die Maße *sp* und *sppf*, so ist unmittelbar ersichtlich, dass eine lineare Skala den Wertebereich nicht darstellen kann, da zwei Ausreißer die Darstellung verzerren. In einer logarithmischen Skala ist jedoch die Tendenz einer Korrelation erkennbar (siehe Abbildung 3b). Auch die Maße *cars* und *fields* weisen eine leichte Aufwärtstendenz auf, doch auch hier lässt die hohe Standardabweichung keinen klaren Schluss zu. Eine genauere Untersuchung der Ausreißer könnte aber lohnend sein.

Ebenso wie bei den verwandten Maßen *avdg* und *avdgog* weisen die Maße *mc* und *mcop* keine nachweisbare Tendenz auf. Das Konzept der *unintuitiven Züge* sollte noch einmal genauer untersucht werden, da die vorliegenden Daten einen leichten Aufwärtstrend suggerieren, jedoch auch hier keine zweifelsfreie Erkenntnis gewonnen werden kann.

**GRUPPIERUNG NACH EDITION** Um auszuschließen, dass die verschiedenen Editionen inhomogen sind und nach verschiedenen Maßstäben entworfen wurden, tragen wir die Durchschnittswerte der Maße über alle Spiele einer Edition auf und erhalten Abbildung 4. Erkennbar ist, dass sich – wie erwartet – die Junioredition von den anderen Editionen absetzt, die übrigen Editionen unter sich aber homogen zu sein scheinen. Jedoch unterstützt die Junioredition einige bereits oben beschriebene Tendenzen: Unter der Annahme, dass die Startkonfigurationen der Junioredition so entworfen sind, dass sie für Erwachsene leicht zu lösen sind, kann die Tatsache, dass der optimale Lösungsweg der Junior-Startkonfigurationen durchschnittlich deutlich kürzer ist, als der aller anderen Editionen, die Hypothese stützen, dass die Länge des Lösungswegs den wesentlichen Faktor darstellt, nach dem die Spieledesigner die Schwierigkeitsstufen kategorisieren. Ebenso unterstützt die Hinzunahme der Junioredition den Trend zur positiven Korrelation zwischen Komplexität und den Maßen *cars* und *fields* sowie die Tendenz zur negativen Korrelation zwischen Komplexität und der *branching complexity br*.

### 3.4.2 Korrelationen der Maße

Nachdem wir die Korrelation der Maße mit der Schwierigkeitsangabe des Herstellers betrachtet haben, möchten wir untersuchen, inwiefern die Maße untereinander korrelieren. Dazu gehen wir folgendermaßen vor: wir sortieren die untersuchten Startkonfigurationen sukzessive nach jedem Maß und speichern in jedem Schritt die jeweilige Reihenfolge. Wir berechnen dann paarweise zwischen den

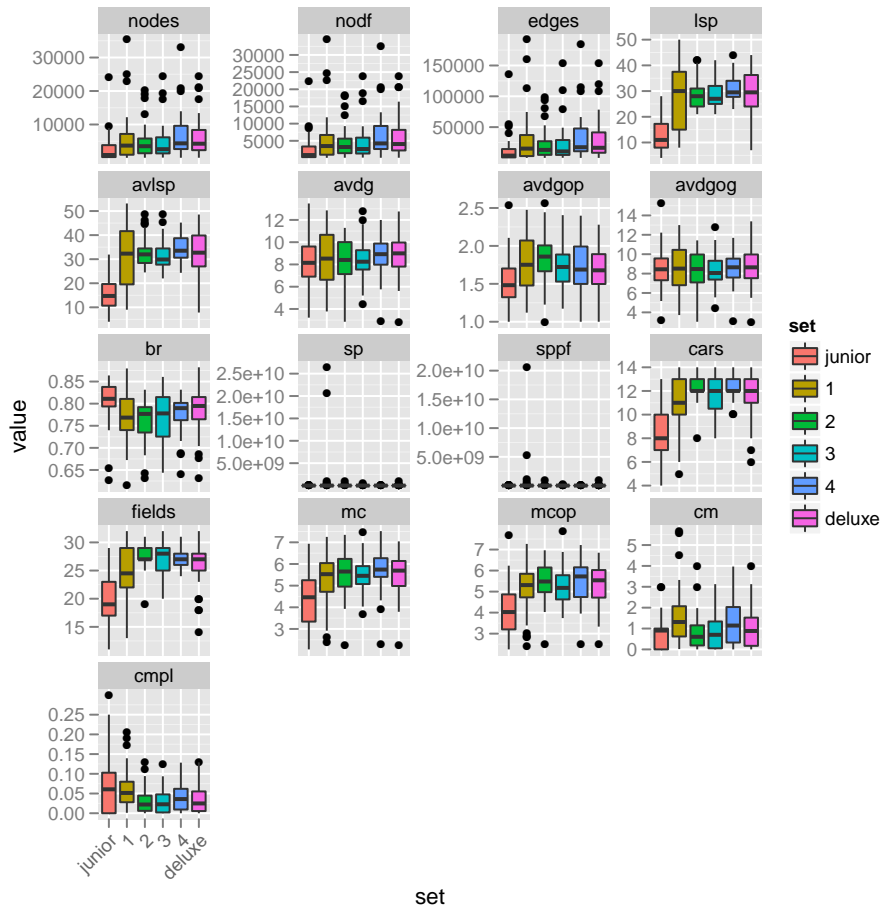


Abbildung 4: Boxplots aller Startkonfigurationen (inklusive Junioredition und Dopplungen über Editionen hinweg)



Sortierungen den Korrelationskoeffizienten nach Pearson. Dieser ist für zwei Zufallsvariablen  $X$  und  $Y$  definiert als

$$\rho = \frac{\text{Cov}(X, Y)}{\sigma(X)\sigma(Y)}$$

wobei  $\text{Cov}$  die Kovarianz von  $X$  und  $Y$  mit  $\text{Cov}(X, Y) = E[XY] - E[X]E[Y]$ ,  $E[\cdot]$  Erwartungswert, und  $\sigma$  die jeweilige Standardabweichung bezeichnet. Ein Korrelationskoeffizient von 1 entspricht einem perfekten positiven linearen Zusammenhang, ein Wert von -1 einem perfekten negativen Zusammenhang. Bei  $\rho = 0$ , hängen die beiden Variablen in keiner linearer Weise voneinander ab.

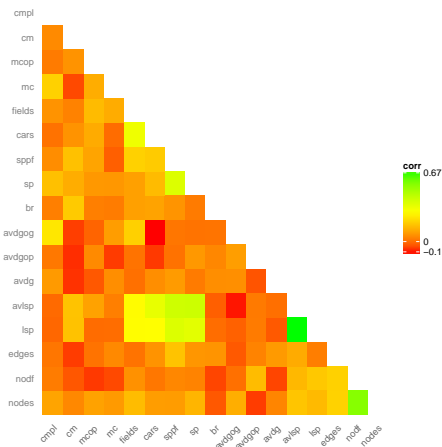
Um die Korrelation zu visualisieren, verwenden wir *heatmaps*, wie in Abbildung 5 zu sehen ist.

In Abbildung 5a ist die Korrelation der 17 Maße untereinander aufgetragen, wenn man alle Startkonfigurationen in die Betrachtung miteinbezieht. Auf den ersten Blick fällt auf, dass der Großteil der Maße in keinerlei Korrelation zueinander steht. Es gibt einige leicht negative Korrelationen, jedoch sind diese zu gering, um als signifikant zu gelten. Korrelationen zwischen den Maßen *nodes* und *nodf*, *cars* und *fields*, *sp* und *sppf* sowie *lsp* und *avlsp* waren zu erwarten, da die Definitionen dieser Maße jeweils sehr ähnlich zueinander sind. Diese sind auch in allen anderen Heatmaps zu erkennen. Die positive Korrelation zwischen den Maßen *lsp/avlsp* und *fields*, *cars*, *sp* und *sppf* stützt unsere vorherige Beobachtung, dass diese Maße eine Rolle in der Beurteilung der Komplexität einer Startkonfiguration durch den Hersteller spielen. Denn in Abbildung 3 ist eine positive Korrelation zwischen Schwierigkeitskategorie des Herstellers und dem Maß *lsp* bereits klar zu erkennen. Die positive Korrelation zwischen *lsp* und den genannten Maßen kann daher als Hinweis dienen, dass auch ein Zusammenhang zwischen der Komplexität und den Maßen *fields*, *cars*, *sp* und *sppf* bestehen könnte.

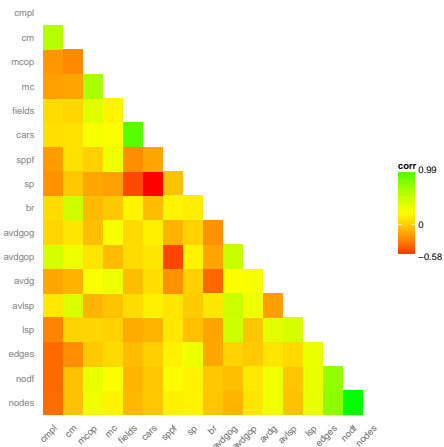
Die fünf weiteren *heatmaps* beinhalten die Korrelation der Maße untereinander, wenn nur die Startkonfigurationen einer Schwierigkeitsstufe betrachtet werden. Zunächst kann festgehalten werden, dass die Korrelationen sowohl in positive als auch in negative Richtung stärker ausgeprägt sind, wenn nur die Startkonfigurationen eines Schwierigkeitsgrades berücksichtigt werden, allerdings treten auch weitere Korrelationen auf, die in Abbildung 5a nicht beobachtet werden konnten. Jedoch gibt es keine auffallend hohe positive oder negative Korrelation, die in allen Schwierigkeitsstufen nachgewiesen werden kann.

### 3.4.3 Clustering der Spiele anhand ihrer Komplexitätsmaße

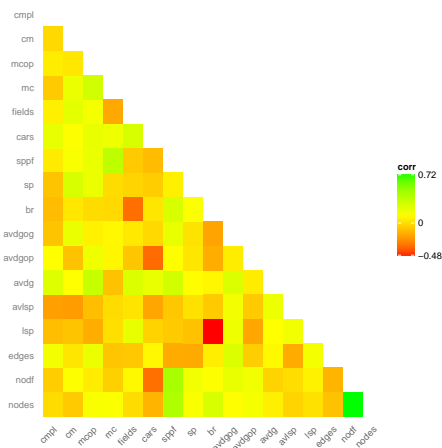
Obwohl die schwache Korrelation zwischen Maßen und Schwierigkeitsangabe darauf hinweist, dass die Startkonfigurationen anhand der betrachteten Maße nicht in die vom Hersteller vorgegebenen Gruppen eingeteilt werden können, benutzen wir als letzten Schritt einen Clusteringalgorithmus um festzustellen, inwiefern unsere Maße es ermöglichen, die vom Hersteller angegebenen Schwierigkeitsstufen der Konfigurationen vorherzusagen. Da die Anzahl der Gruppen (fünf, eine für jede Schwierigkeitsstufe) im Voraus bekannt ist, bietet sich ein *k-means-Algorithmus* an, der in unserem Fall 213 17-dimensionale Datenpunkte in  $k = 5$



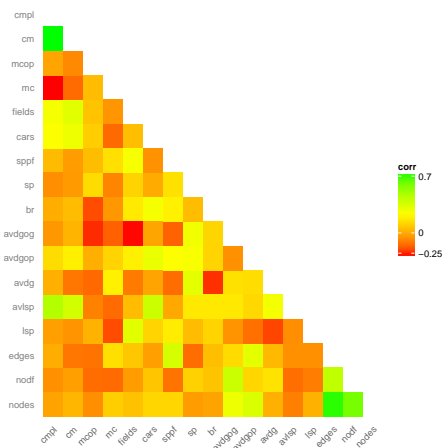
(a) Alle Startkonfigurationen



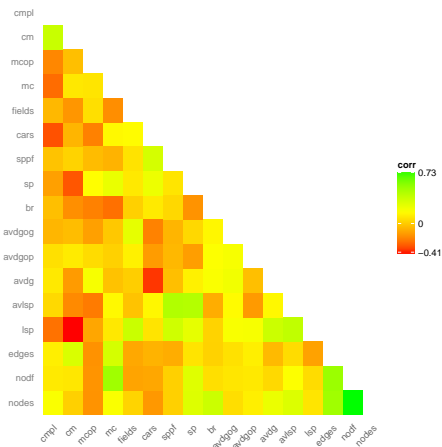
(b) *beginner*



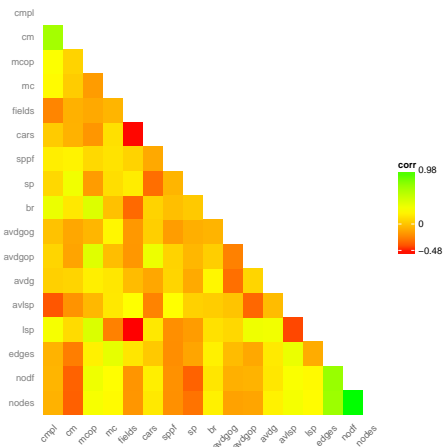
(c) *intermediate*



(d) *advanced*



(e) *expert*



(f) *grand master*

Abbildung 5: Heatmaps zur Visualisierung der Korrelationen zwischen den einzelnen Maßen (jeweils ohne die Startkonfigurationen der Junioredition)

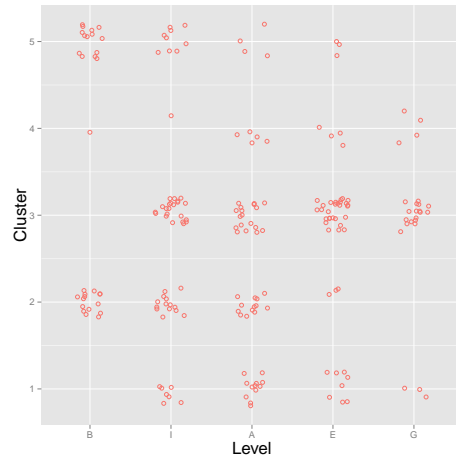


Abbildung 6: Ergebnis des *k-means-clustering* auf Basis der errechneten Maße. Jeder Punkt stellt eine Startkonfiguration dar, auf der  $x$ -Achse ist ihre Schwierigkeitsstufe, auf der  $y$ -Achse ist ihre Einordnung in die Gruppen aufgetragen. Es handelt sich der besseren Darstellung halber um einen Jitter-Plot, eigentlich liegen alle Punkte auf den Kreuzungspunkten des Gitters.

Cluster einteilt, sodass innerhalb der Cluster die Summe der Quadrate minimiert wird [10]. Als Distanzfunktion wird die euklidische Distanz zwischen zwei Punkten im 17-dimensionalen durch die Maße definierten Vektorraum verwendet. Der benutzte Algorithmus prüft in jedem Iterationsschritt für jeden Punkt, welchem existierenden Schwerpunkt er am nächsten liegt und ordnet ihn dem entsprechenden Cluster zu. Dann werden die Schwerpunkte der neuen Cluster berechnet. Der Algorithmus terminiert entweder, wenn eine bestimmte Iterationstiefe erreicht ist oder wenn die Schwerpunkte und Zuordnungen konstant bleiben.

Um die unterschiedlichen Größenordnungen der Maße auszugleichen, normieren wir alle Maße auf das Intervall  $[0, 1]$ . Wir erhalten die Zuordnung jeder Startkonfiguration in eine von fünf Gruppen, die in Abbildung 6 zu sehen ist. Deutlich ist zu erkennen, dass ein *k-means-clustering* anhand der vorliegenden Maße nicht in der Lage ist, die vom Hersteller vorgeschlagene Einteilung vorzunehmen. Denn ausnahmslos alle enthalten nahezu die gleiche Anzahl von Spielen aller Schwierigkeitsgrade. Allerdings ist dennoch eine leichte Tendenz zu erkennen, dass Cluster entweder *beginner-* oder *grand master-*Spiele enthalten. Allerdings kann festgehalten werden, dass die vorgestellten Maße als Gesamtheit nicht geeignet sind, um die betrachteten Startkonfigurationen in die vom Hersteller vorgeschlagenen Kategorien einzuteilen. Nun ist die Frage, ob unsere Maße mit der von Menschen empfundenen Komplexität besser korrelieren. Um dies zu testen, schlagen wir ein Experiment vor, dessen Konzeption, Durchführung und Ergebnisse wir in Kapitel 4 beschreiben.



# 4

## EXPERIMENT

### 4.1 KONZEPTION: AUFBAU UND DURCHFÜHRUNG DES EXPERIMENTS

Wir haben in Kapitel 3 gezeigt, dass die Hersteller keine ganz klare Regel hatten, nach der sie die Spiele in Schwierigkeitsstufen eingeteilt haben, konnten diese Regel jedoch auch noch nicht quantifizieren. Nun fragen wir uns, ob die Einteilung der Schwierigkeit mit der im informatischen Sinne verstandenen Komplexität – also der Anzahl der Lösungsschritte – korreliert, und ob wir eine Korrelation zwischen den Lösungszeiten und einem unserer Maße nachweisen können. Dazu schlagen wir mit Erlaubnis des Herstellers ein Experiment vor, das durch Versuchsdaten mehr Klarheit in die Fragestellungen bringen soll. Wir haben ein Java-Applet erstellt, das es Versuchspersonen erlaubt, *Rush Hour* zu spielen und das die dabei durchgeführten Spielzüge für eine Analyse speichert. Wir haben das Applet als *html-object* in unsere Website eingebunden, sodass es den Versuchspersonen möglich war, im Browser eines beliebigen Rechners an dem Experiment teilzunehmen.

In Abbildung 7 ist die Oberfläche des Applets zu sehen. Erkennbar ist, dass den Spielern angezeigt wird, wieviele Spiele sie bereits gespielt und wieviele sie bereits gelöst haben sowie die Anzahl der verbliebenen Spiele in der Testreihe. Des weiteren können sie die Anzahl ihrer im aktuellen Level bereits getätigten Züge sehen. Es ist möglich, einen Zug rückgängig zu machen sowie das Level abubrechen. Es ist explizit keine Möglichkeit gegeben, ein Level neu zu starten oder mehrfach zu spielen. Wir haben die Versuchspersonen in unserer Anleitung angewiesen [1], nach Möglichkeit jedes Level zu Ende zu spielen und es nur im Ausnahmefall abubrechen. Zudem haben wir sie gebeten, eine möglichst kurze Lösung zu finden, also mit möglichst wenigen Zügen einen Endzustand zu erreichen. Da die Spieler nicht unter unserer direkten Aufsicht stehen – ja sogar die Angabe ihres Namens optional ist –, mussten

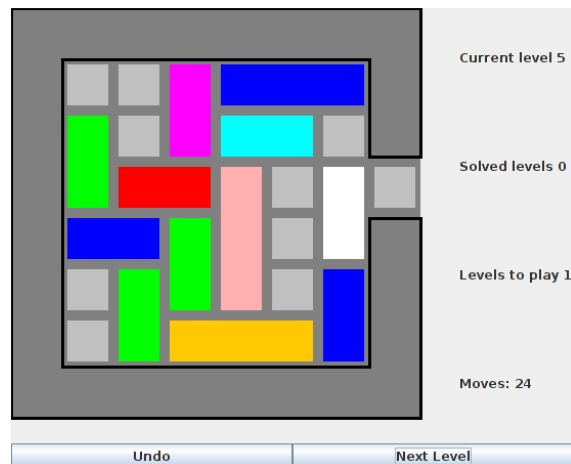


Abbildung 7: Das Design unseres Applets

wir ein Mittel finden, sowohl nicht ernsthafte Spieler auszuschließen als auch einen Anreiz zu schaffen, Level möglichst schnell und zu Ende zu spielen. Dies haben wir einerseits durch gezielte Auswahl der Versuchspersonen, andererseits durch die Ankündigung gelöst, dass die besten drei Spieler einen Einkaufsgutschein erhalten. Um Spieler, die bereits ein Level nicht lösen konnten, zum Weiterspielen anzuregen, obwohl sie vermutlich keine Möglichkeit mehr haben, den Einkaufsgutschein zu gewinnen, haben wir weiterhin angekündigt, dass unter den übrigen Spielern zusätzlich sechs Einkaufsgutscheine niedrigeren Werts verlost werden, wobei die Gewinnchancen steigen, je mehr Level erfolgreich beendet wurden.

Da die Aussicht auf Geldgewinne jedoch potentiell Spieler anlockt, die entweder mit unehrlichen Mitteln spielen oder viel Zeit investieren, um zu gewinnen, mussten wir ausschließen, dass Versuchspersonen die Möglichkeit gegeben ist, Level mehrfach zu spielen und erst nach mehrmaligem Üben an unserem Gewinnspiel teilzunehmen. Dies würde unsere Versuchsdaten verfälschen und potentiell unbrauchbar machen, weshalb wir uns für die Einführung von Zugangscodes entschieden haben: jeder Versuchsteilnehmer erhält im Vorhinein (per Email oder persönliche Vergabe) einen eindeutigen Zugangscodes, der korrekt eingegeben werden muss, damit das Applet startet. Es wird dann gespeichert, wie oft jeder Zugangscodes bereits eingegeben wurde. Damit ist ausgeschlossen, dass wir wiederholte Spiele bei der Analyse betrachten. Bevor eine Versuchsperson mit dem eigentlichen Spiel beginnt, bitten wir sie einige Fragen zu sich zu beantworten. Neben diesen Informationen speichern wir jeden Zug der Versuchsperson zusammen mit einem Zeitstempel. Zudem bitten wir die Spieler nach jedem erfolgreich beendeten Level um eine Einschätzung der Schwierigkeit dieses Levels. Sie können das Spiel dann auf einer Skala von 1 (sehr einfach) bis 5 (sehr schwer) bewerten.

### *Selektion der Level*

Wir haben aus der uns verfügbaren Menge der Startkonfigurationen 24 ausgewählt, die sich hinsichtlich der Schwierigkeitsangabe des Herstellers unterscheiden, um sie näher zu untersuchen. Um den Zeitaufwand in einem für die Versuchspersonen zumutbaren Rahmen zu halten, wurden diese Startkonfigurationen in vier Teilmengen unterteilt, die jeweils zwei *beginner*-, ein *intermediate*-, ein *advanced*-, ein *expert*- (in einer Teilmenge zwei) und ein *grand master*-Konfigurationen (in einer Teilmenge eins) in dieser Reihenfolge enthalten, also insgesamt sechs Level. Die Idee ist, dass für jede Versuchsperson eine eindeutige, zufällige Reihenfolge der vier Levelsets erzeugt wird, sodass ihr bei jedem „Neustart“ sechs neue Level zur Aufgabe gegeben werden. Dies kann durch die Zugangscodes gewährleistet werden: für jeden eingegebenen Zugangscodes wird eine zufällige Permutation der vier Levelmengen erstellt und bei der *i*. Eingabe desselben Zugangscodes das *i*. Levelset der Permutation als Spielset verwendet. Da die Zuordnung der Zugangscodes auf die Permutationen gleichverteilt gewählt ist, ist sowohl gewährleistet, dass alle ausgewählten Level annähernd gleich oft gespielt werden, als auch dass jede Versuchsperson jedes Level höchstens einmal spielen kann (vorausgesetzt sie hat genau einen Zugangscodes zur Verfügung). Weiterhin ist es so für alle möglich, mehrere Levelsets zu spielen. Mögliche Lerneffekte halten wir in diesem Szenario

für zu gering, als dass sie berücksichtigt werden müssten – zumal wir keinerlei Anforderungen an die Vorkenntnisse der Versuchsteilnehmer stellen, sodass davon auszugehen ist, dass diese ohnehin unterschiedlich sind.

In Tabelle 4 sind die einzelnen Maße der 24 ausgewählten Startkonfigurationen zu sehen sowie die jeweiligen Mittelwerte und Standardabweichungen innerhalb der Schwierigkeitskategorien. Rot markiert sind die Werte, aufgrund derer wir diese Level ausgewählt haben. Wir haben bewusst Startkonfigurationen ausgewählt, die in einigen Maßen Ausnahmen in ihrer Kategorie sind, sodass wir den Einfluss dieses Maßes auf die Komplexität untersuchen können.

Unter den *beginner*-Konfigurationen finden wir ein Spiel mit einem außergewöhnlich großem Zustandsraum: die Hypothese ist, dass dies ein Faktor für eine höhere empfundene Komplexität ist, da Versuchsteilnehmer einen größeren Raum explorieren müssen, um einen Lösungszustand zu finden, oder ob Versuchsteilnehmer die Größe des Zustandsraumes gar nicht wahrnehmen. Daneben haben wir ein *beginner*-Spiel mit niedrigem Durchschnittsknotengrad und ansonsten durchschnittlichen Werten ausgewählt, mit dem wir uns Antwort auf die Frage erhoffen, ob die durchschnittliche Anzahl der Wahlmöglichkeiten pro Zug die empfundene Schwierigkeit beeinflusst.

Für die Zusammenstellung der *intermediate*-Spiele haben wir neben einigen in allen Maßen durchschnittlichen Spielen ein Spiel mit auffallend niedriger Anzahl an benötigten Zügen ausgewählt sowie ein Spiel mit auffallend hoher Anzahl an benötigten und unintuitiven Zügen.

Unter den *advanced*-Spielen haben wir ebenfalls zwei durchschnittliche und zwei Level mit außergewöhnlichen Werten gewählt: eines hat einen für seine Kategorie niedrigen Durchschnittsknotengrad, das zweite ist in vielen Maßen ungewöhnlich. Neben einer hohen Anzahl an nötigen Zügen hat es ebenfalls einen niedrigen durchschnittlichen Knotengrad sowie einen hohen Wert an unintuitiven Zügen neben einer ungewöhnlich hohen Anzahl an Autos auf dem Spielbrett.

Für die *expert*-Spiele haben wir ein Spiel ausgewählt, das selbst im Vergleich mit anderen Kategorien über einen sehr niedrigen Knotengrad verfügt und das das einzige Spiel ist, das genau einen optimalen Weg besitzt. Trotzdem wird dieses Spiel von den Herstellern als *expert* eingestuft. Wir fragen uns daher: Hat dieses Spiel weitere Eigenschaften, sodass es auch von Versuchsteilnehmern als ähnlich schwer empfunden wird? Im Gegensatz dazu haben wir ein weiteres Spiel derselben Kategorie ausgewählt, das einen großen Zustandsraum, viele optimale Wege und einen relativ hohen Knotengrad besitzt. Werden diese beiden Spiele trotz der fast gegensätzlichen Eigenschaften von den Spielern als ähnlich schwer empfunden?

Unter den *grand master*-Spielen finden wir zwei mit durchschnittlichen Werten und eins mit einem außergewöhnlich hohen Wert für die (gewichtete) Anzahl an unintuitiven Züge.

#### 4.2 VERSUCHSERGEBNISSE

Für das Experiment haben wir das Spiel über einen Zeitraum von 10 Tagen freigeschaltet und in verschiedenen Umfeldern Versuchsteilnehmer gesucht. Der Großteil der Versuchsteilnehmer ist jedoch in die Gruppen der Studierenden der

	NODES	EDGES	LSP	AVDG	BR	SP	SPPF	CARS	FIELDS	MC	MCOP	CM	CMPL
<i>Mean</i>	5245	25434	26.4	8.6	0.78	$20.77 \cdot 10^7$	$17.98 \cdot 10^7$	11.26	25.5	5.32	5.1	1.08	0.04
<i>SD</i>	5882	32628	9.96	1.95	0.05	$210.7 \cdot 10^7$	$134 \cdot 10^7$	2.14	4.2	1.09	1.08	1.05	0.05
<b>B</b>													
<i>Mean</i>	3613	19419	10.1	9.32	0.8	989.1	174.0	8.6	19.9	4.89	4.77	0.64	0.06
<i>SD</i>	4646	26817	3.7	2.4	0.06	2494	309	2.6	5.25	1.41	1.48	0.69	0.08
del <sub>07</sub> B	7273	41384	12	12.1	0.88	576	18	9	18	6.6	6.7	1.1667	0.0972
1 <sub>03</sub> B	4821	25166	15	10.5	0.85	1536	384	10	21	6.43	6.06	0.9167	0.0611
del <sub>01</sub> B	1075	5821	7	10.9	0.86	61	15.3	8	20	5.4	5.8	0	0
del <sub>04</sub> B	451	2008	8	9.07	0.8	2	2	7	18	3.8	3.3	1	0.125
del <sub>05</sub> B	2784	14786	8	10.66	0.8	575	288	11	26	6.0	6.2	0	0
del <sub>02</sub> B	21055	119889	7	11.5	0.8	42	42	11	25	6.5	6.85	0	0
del <sub>06</sub> B	2954	14047	8	9.58	0.87	6	6	11	26	5.48	6.25	1	0.13
1 <sub>10</sub> B	51	94	14	3.78	0.61	8	8	11	24	2.9	2.8	1.25	0.09
<b>I</b>													
<i>Mean</i>	4139	1806	21.5	8.3	0.78	$0.271 \cdot 10^7$	$0.135 \cdot 10^7$	11.0	25.22	5.2	5.0	0.75	0.04
<i>SD</i>	4364	25978	7.59	1.99	0.053	$1.103 \cdot 10^7$	$0.798 \cdot 10^7$	2.37	4.70	1.11	1.15	0.82	0.048
2 <sub>07</sub> I	3182	13013	24	8.83	0.78	$0.019 \cdot 10^7$	63840	12	28	5.65	5.79	0.3357	0.0140
3 <sub>08</sub> I	1338	4786	26	7.26	0.73	48840	16280	14	31	5.82	6.12	0.0020	0.0001
1 <sub>15</sub> I	1128	3751	14	6.70	0.77	228	228	14	32	5.67	4.85	1.5000	0.1071
1 <sub>19</sub> I	561	1604	40	5.90	0.79	123	24.6	11	25	3.7	3.96	5.5772	0.1394
<b>A</b>													
<i>Mean</i>	5562	27910	25.8	8.9	0.78	$3.01 \cdot 10^7$	$2.38 \cdot 10^7$	11.3	25.7	5.4	5.1	0.87	0.03
<i>SD</i>	5932	32968	6.68	1.96	0.053	$15.9 \cdot 10^7$	$15.1 \cdot 10^7$	1.81	3.48	0.99	1.01	1.0	0.04
1 <sub>26</sub> A	196	483	22	5.0	0.82	8	8	9	20	2.64	2.88	4.5000	0.2045
1 <sub>23</sub> A	4671	19068	29	8.58	0.74	3782	3782	14	30	6.12	5.08	1.4804	0.0510
4 <sub>18</sub> A	8052	40360	29	10.2	0.79	$0.21 \cdot 10^7$	21480	10	24	5.6	5.4	0.6562	0.0226
1 <sub>22</sub> A	530	1514	33	5.9	0.73	468	78	14	31	4.73	4.65	5.6667	0.1717
<b>E</b>													
<i>Mean</i>	6213	29350	31	8.27	0.77	$105.5 \cdot 10^7$	5.84	11.98	26.96	5.46	5.29	0.95	0.033
<i>SD</i>	7780	41649	7.26	1.85	0.05	$487.8 \cdot 10^7$	$311.3 \cdot 10^7$	1.79	3.37	1.22	1.11	1.17	0.04
del <sub>39</sub> E	754	2231	32	6.0	0.71	21600	7200	12	27	4.39	4.10	0.0125	0.0004
2 <sub>28</sub> E	75	106	31	2.85	0.63	1	1	13	29	2.26	2.52	4.0000	0.1290
3 <sub>25</sub> E	6262	30793	29	10.41	0.82	18240	3040	11	27	5.98	5.58	0.9000	0.0310
1 <sub>32</sub> E	23009	113755	48	10.02	0.80	$2058 \cdot 10^7$	$2058 \cdot 10^7$	13	29	6.94	5.97	2.3959	0.0499
1 <sub>38</sub> E	3493	13866	50	8.02	0.70	$12.7 \cdot 10^7$	$2.1 \cdot 10^7$	13	29	5.74	6.66	0.2652	0.0053
<b>G</b>													
<i>Mean</i>	6380	29749	37.3	8.44	0.79	$4.1 \cdot 10^7$	$4.0 \cdot 10^7$	11.88	26.63	5.37	5.28	0.98	0.03
<i>SD</i>	6293	35888	3.18	1.54	0.03	$18.8 \cdot 10^7$	$18.8 \cdot 10^7$	1.08	1.79	0.72	0.75	0.79	0.02
2 <sub>37</sub> G	5824	23352	41	8.21	0.78	$0.11 \cdot 10^7$	276080	11	25	5.05	5.20	0.8796	0.0215
del <sub>55</sub> G	1583	5129	41	6.78	0.77	2688	448	11	25	4.32	3.95	2.5268	0.0616
2 <sub>36</sub> G	1062	2992	38	5.76	0.75	12320	3080	12	26	3.93	4.03	0.1312	0.0035

Tabelle 4: Eine Übersicht über die Maße der für das Experiment ausgewählten Level. Markante Werte, die als Basis für die Auswahl dienten, sind rot markiert. Mittelwert (*mean*) und Standardabweichung (*SD*) wird pro Schwierigkeitsgruppe und insgesamt angezeigt.



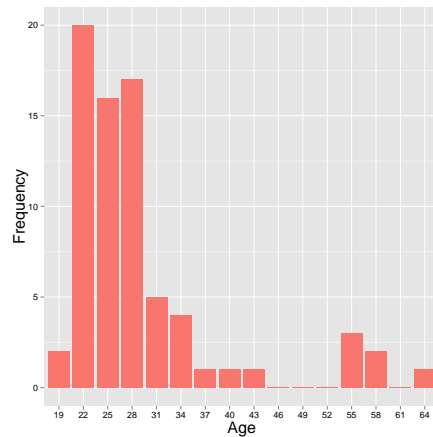
Mathematik, Informatik und Physik einzuordnen. Nicht alle Teilnehmer haben alle sechs Level einer Runde zu Ende gespielt, für die folgende Analyse werden nur die Daten der Spiele verwendet, die beendet wurden. Weitere 31 Spiele wurden nicht beendet und kommen nicht zur Verwendung. Einige Teilnehmer haben die Möglichkeit genutzt, mehrere Runden mit verschiedenen Levels spielen zu können, wodurch wir 97 Datensätze mit jeweils sechs gespielten Levels, für die Analyse verwenden konnten. Während des Experiments ist mindestens ein Fehler innerhalb des Applets aufgetreten, der die Kollisionsbehandlung der Autos betrifft. Dieser ist vermutlich – da er nicht reproduzierbar war – auf die *Java Runtime Environment* des entsprechenden Clients zurückzuführen. Um weitere potentiell fehlerhafte Versuchsdaten von der weiteren Analyse auszuschließen, haben wir alle Versuchsdaten zunächst einer Validitätsprüfung unterzogen. In Tabelle 5 ist zu sehen, welche Level von wievielen Spielern gespielt und gelöst wurden, nachdem alle fehlerhaften und nicht beendeten Level aus den Versuchsdaten entfernt wurden.

	DELUXE 07B	1 03B	DELUXE 01B	DELUXE 04B	DELUXE 05B	DELUXE 02B
gespielt	23	23	20	21	20	20
gelöst	23	23	20	20	20	20
	DELUXE 06B	1 10B	2 07I	3 08I	1 15I	1 19I
gespielt	29	29	22	21	21	29
gelöst	27	26	17	18	20	18
	1 26A	1 23A	4 18A	1 22A	DELUXE 39E	2 28E
gespielt	22	20	20	28	22	21
gelöst	18	13	15	22	15	15
	3 25E	1 32E	1 38E	2 37G	DELUXE 55G	2 36G
gespielt	20	29	30	22	21	21
gelöst	20	17	9	10	14	13

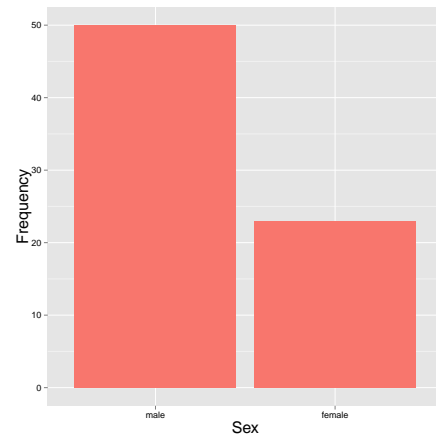
Tabelle 5: Eine Übersicht der Versuchsergebnisse: Anzahl der Personen, die die jeweiligen Level gespielt und gelöst haben. Hier sind schon diejenigen Daten ausgenommen, die aus unvollständigen Spielen stammen.

#### 4.2.1 Spieler

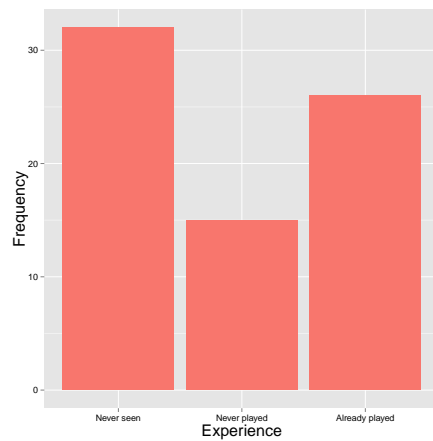
Wir konnten 74 Spieler gewinnen, an unserem Experiment teilzunehmen, eine Übersicht über die Spielerinformationen kann Abbildung 8 entnommen werden. Man sieht, dass die meisten Spieler zwischen 20 und 30 Jahre alt sind, doppelt so viele Männer wie Frauen teilgenommen haben und der Großteil der Spieler *Rush Hour* noch nie gespielt zuvor oder sogar noch nie gesehen hat. Die Antwort, *Rush Hour* oft zu spielen, ist von keinem Spieler gegeben worden. Somit kann näherungsweise davon ausgegangen werden, dass alle Spieler ungefähr die gleichen Voraussetzungen mitbringen. In Abbildung 8d ist zu erkennen, dass ein



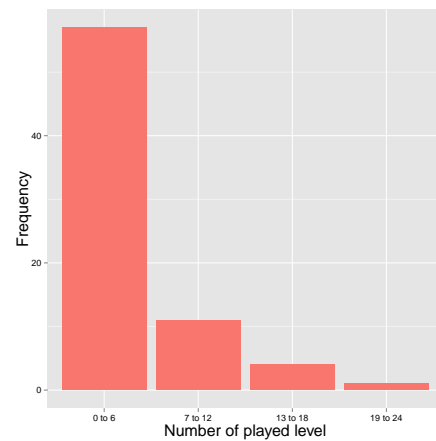
(a) Ein Histogramm zur Darstellung der Altersverteilung der Versuchsteilnehmer



(b) Ein Histogramm zur Darstellung der Geschlechterverteilung der Versuchsteilnehmer



(c) Ein Histogramm zur Darstellung der Vorkenntnisse der Versuchsteilnehmer, was das Spiel *Rush Hour* betrifft



(d) Ein Histogramm, das darstellt, wieviele Level die Versuchspersonen gespielt haben

Abbildung 8: Eine Übersicht über unsere Informationen über die Versuchsteilnehmer. Da die Angaben der Spieler nicht kontrolliert werden konnten, können wir diese Angaben nicht garantieren.

Großteil der Spieler eine Runde gespielt hat, jedoch eine zweistellige Anzahl an Spielern mehrere Runden gespielt hat.

#### 4.2.2 Abgebrochene und gelöste Level

Zunächst untersuchen wir, welche Level von unseren Versuchspersonen gelöst und welche abgebrochen wurden. Absolute Zahlen sind Tabelle 5 zu entnehmen, eine genauere Aufschlüsselung ist in Abbildung 9 zu sehen. Gegen die einzelnen Level aufgetragen ist die relative Anzahl der Spieler, die dieses Level gelöst, optimal gelöst oder abgebrochen haben. Die Level auf der  $x$ -Achse sind nach Schwierigkeitsangabe des Herstellers sortiert.

Wie erwartet ist ein Trend zu erkennen, dass mit aufsteigender Hersteller-schwierigkeitsangabe die Abbrecherquote deutlich steigt. Per Konstruktion sind die Plots der abgebrochenen und gelösten Spiele spiegel-symmetrisch und ergänzen sich zu 100%. Deutlich zu sehen ist, dass es kein einziges Spiel gibt, das von mehr als 60% der Spieler optimal gelöst wurde, selbst bei den vermeintlich leichten Spielen wird ein Wert von 50% nur dreimal erreicht. Ab den *inter-*

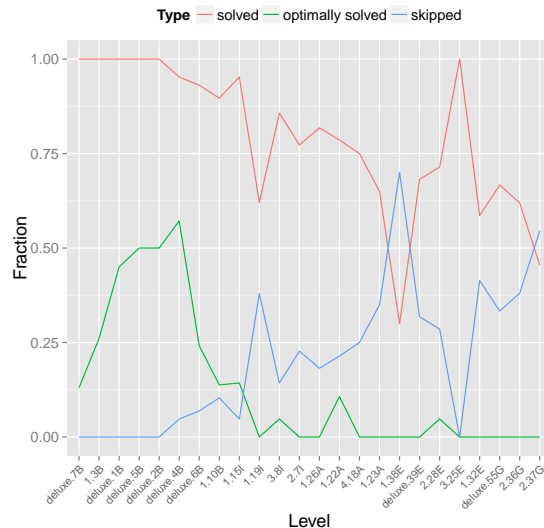


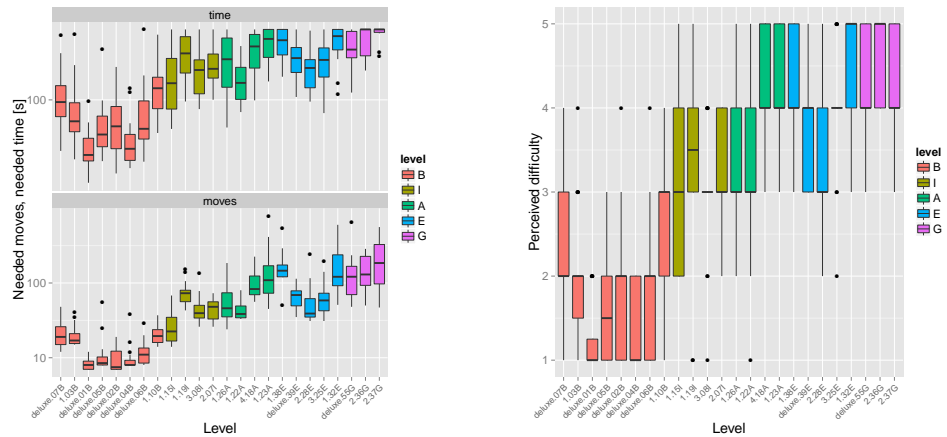
Abbildung 9: Anteil der gelösten, optimal gelösten und abgebrochenen Level

*mediate*-Spielen schafft es nur noch eine auffallend geringe Zahl, einen optimalen Lösungsweg zu finden. Auch die Plots der abgebrochenen und optimal gelösten Level zeigen eine gewisse Symmetrie. Während bei den *beginner*-Spielen kaum eine Versuchsperson das Level abbricht, es jedoch einige schaffen, es optimal zu lösen, steigt die Abbrecherquote ab dem Level 1 15I an und bleibt ungefähr bei 30%. Gleichzeitig ist der Anteil der optimalen Lösungen bis auf ein paar Ausnahmen, die die 15% nicht überschreiten, bei 0.

Es stellt sich die Frage, welche Level hier besonders auffallen. Bemerkenswert ist, dass das Level *deluxe 02B*, das wir wegen seines großen Zustandsraumes ausgewählt haben, von einem auffallend hohen Anteil Spieler optimal gelöst wurde und von keiner einzigen Versuchsperson abgebrochen wurde. Dies lässt vermuten, dass ein großer Zustandsraum allein kein Kriterium für Komplexität ist. Das *beginner*-Spiel 1 10B, das einen ungewöhnlich niedrigen Knotengrad aufweist, hat innerhalb der *beginner*-Level eine auffallend hohe Abbrecherquote, wohingegen das Spiel 1 03B mit durchschnittlichem Knotengrad, aber ähnlich langem Lösungsweg wie 1 10B, eine deutlich höhere Erfolgsquote hat. Gleichzeitig weist auch das Spiel 2 28E, das ebenfalls wegen seines niedrigen Knotengrades ausgewählt wurde, eine hohe, jedoch innerhalb der *expert*-Spiele nicht außergewöhnliche Abbrecherquote auf. Dies könnte ein Hinweis sein, dass ein niedriger Knotengrad einen Beitrag zur Komplexität leistet. Dies muss jedoch in weiteren Analysen geprüft werden.

Genauer betrachtet werden sollte außerdem das Spiel 3 25E, das als einziges der fortgeschrittenen Spiele und als eines von insgesamt sechs Spielen von allen Spielern gelöst wurde. Dieses ist jedoch hinsichtlich unserer Maße nicht außergewöhnlich. Das Spiel mit der mit Abstand am höchsten Abbrecherquote ist bemerkenswerterweise kein Spiel, das vom Hersteller als *grand master* klassifiziert wurde, sondern ein *expert*-Spiel: der minimale Lösungsweg des Levels 1 38E umfasst allerdings außergewöhnlich viele Schritte, was den hohen Anteil der Spieler, die dieses Level abgebrochen haben, erklärt.

Dass sowohl die Spiele 1 26A und 1 22A als auch das Spiel 2 28E mit durchschnittlich langen Lösungswegen, aber einem ungewöhnlich hohen Wert an unintuitiven



(a) Eine Übersicht über die Lösungslänge und Lösungszeit in Sekunden der Spieler für die einzelnen Level. Die Level sind nach Schwierigkeitsklassifikation des Herstellers gefärbt. Man beachte die logarithmische Skala auf der y-Achse.

(b) Eine Übersicht über die Schwierigkeitseinschätzungen der Spieler, nachdem sie das jeweilige Level beendet haben. Man beachte, dass abgebrochene Level hier nicht berücksichtigt werden.

Abbildung 10: Eine Übersicht über die Lösungen der Spieler: auf der  $x$ -Achse sind die Level, sortiert nach Schwierigkeitskategorie des Herstellers, auf der  $y$ -Achse die benötigte Zeit, die benötigten Züge oder die nach Beenden des Spiels angegebene Schwierigkeit des Levels. Hier sind nur die gelösten Spiele enthalten.

Zügen, eine niedrigere Abbrecherquote oder einen höheren Anteil optimal gelöster Spiele aufweisen, widerspricht zunächst unserer Hypothese, dass unintuitive Züge ein Spiel komplexer machen, jedoch kann auf Basis der vorliegenden Daten noch nicht von ihrer Widerlegung gesprochen werden.

#### 4.2.3 Lösungen der Spieler

Wir untersuchen im Folgenden die von Spielern gelösten Level genauer, wofür wir zunächst die absolute Anzahl an Zügen und die benötigte Zeit betrachten, die die Spieler benötigt haben, um das Spiel zu lösen. Hier und in den folgenden Analysen, bei denen wir die benötigte Zeit zum Lösen der Level verwenden, gehen wir davon aus, dass ein Spieler nicht aktiv spielt, wenn er in einem Spielzustand länger als 10 Minuten verweilt. Aus diesem Grund haben wir in unserer Analyse die Überlegungszeiten in einzelnen Spielzuständen auf maximal 10 Minuten begrenzt und in eventuellen Durchschnittsbildungen und Summierungen über alle Zustände eines Spiels die derart begrenzten Werte verwendet. Zudem können wir die Einschätzungen der Spieler über die empfundene Schwierigkeit der Level als Vergleichskriterium heranziehen. In [Abbildung 10](#) sind diese Informationen als Boxplots dargestellt. Dort sind die Level auf der  $x$ -Achse aufgetragen, nach Schwierigkeitsangabe des Herstellers sortiert.

Auf den ersten Blick ist ersichtlich, dass der Zusammenhang zwischen Schwierigkeitsangabe des Herstellers und der Lösungslänge bzw. -zeit nur ungefähr besteht. Dennoch sollten diese Plots einmal genauer betrachtet werden. Wir beobachten zunächst, dass es in allen drei Darstellungen Spiele gibt, die sich

nicht in das Muster der strikt aufsteigenden Schwierigkeit nach Herstellerangaben einordnen lassen. Wir sehen, dass das Spiel 1 10B unter allen *beginner*-Spielen als das schwierigste gilt und sowohl am meisten Züge als auch am meisten Zeit zur Lösung benötigt hat. Allein durch die Länge des Lösungsweges lässt sich dies nicht erklären, da das *beginner*-Level 1 03B einen längeren Lösungsweg benötigt, zwar ähnlich viele Züge hat, aber weniger Zeit benötigt und auch als weniger schwer beurteilt wurde.

Interessant ist, dass die beiden Spiele 1 23A und deluxe 55G, die beinahe hintereinander gespielt wurden, zwei Schwierigkeitskategorien auseinander liegen, sich trotzdem in allen drei Darstellungen sehr ähneln. Dabei unterscheiden sie sich in fast allen Maßen erheblich. Dies lässt sich durch unsere Maße also nicht erklären.

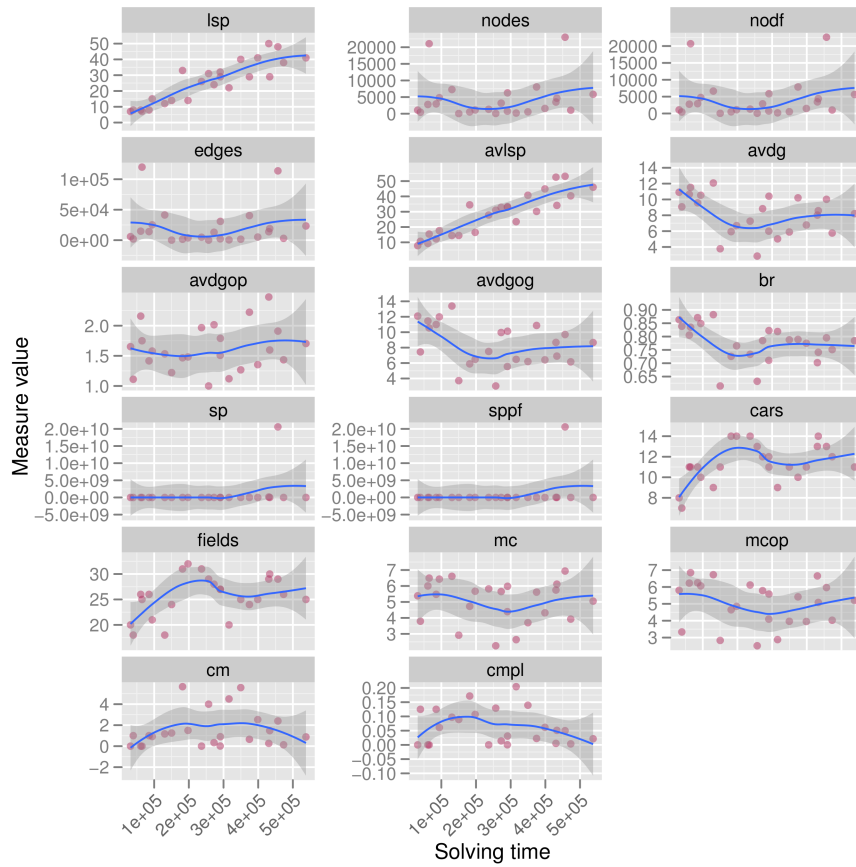
Weiterhin fällt auf, dass bei Level 1 15I die Streuung der Schwierigkeitseinschätzung außergewöhnlich groß ist, obwohl die Streuung dieses Levels in der Zeit- und Zugdarstellung nicht auffällig ist. Betrachtet man die Maße dieses Levels, so bemerkt man, dass dieses Level eine unterdurchschnittliche Anzahl an Zügen im optimalen Lösungsweg aufweist, es aber dennoch von einigen Spielern als schwer empfunden wurde.

#### 4.2.4 Korrelation zwischen Maßen und Versuchsdaten

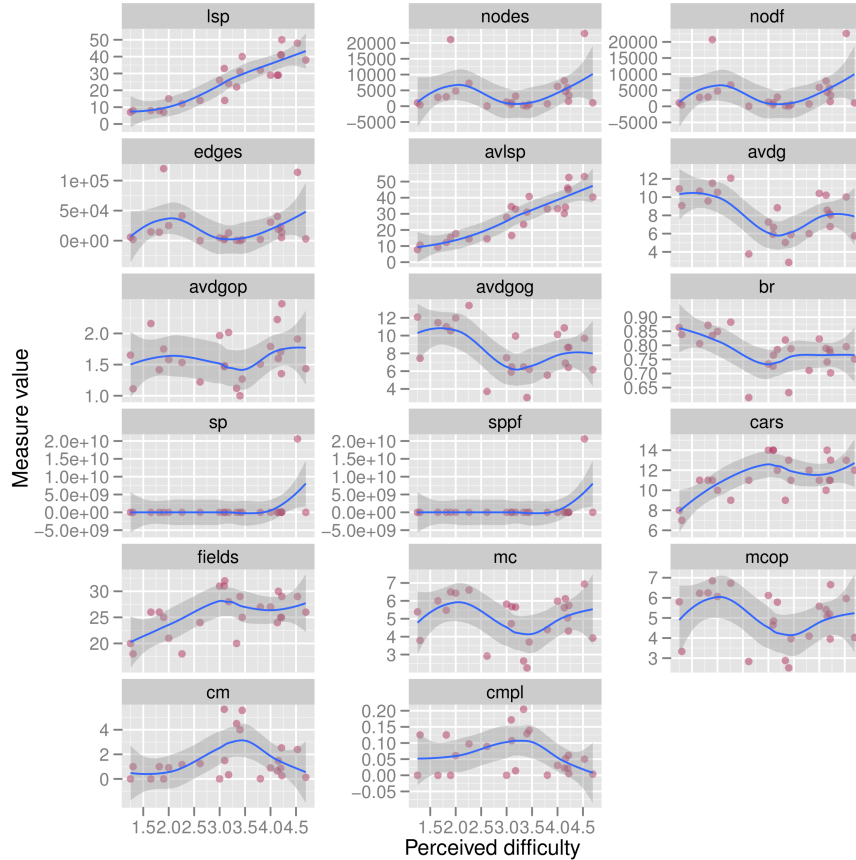
Um systematischer den Zusammenhang zwischen unseren in Kapitel 3 vorgeschlagenen Maßen und den Versuchsergebnissen zu untersuchen, betrachten wir alle gelösten Spiele unseres Experiments und den Zusammenhang zwischen jedem Maß und der zur Lösung benötigten Zeit sowie der von den Spielern angegebenen Schwierigkeit. Wir benutzen diese beiden Eigenschaften, um sowohl auf ein exaktes und präzise quantifizierbares Maß als auch auf eine ungefähre Quantifizierung des empfundenen Aufwands zugreifen zu können. Wir haben diesen eventuellen Zusammenhang in Abbildung 11 visualisiert. Auf der  $x$ -Achse ist in Abbildung 11a die benötigte Zeit zum Lösen dieses Levels als Durchschnitt über alle Spieler aufgetragen, auf der  $y$ -Achse der Wert des jeweiligen Maßes. In den Durchschnitt gehen Spielzustände, bei denen ein Spieler länger als 10 Minuten untätig war, nur mit 10 Minuten ein. In Abbildung 11b ist auf der  $x$ -Achse der Durchschnitt über alle Schwierigkeitseinschätzungen der Spieler aufgetragen. Um einen eventuellen Trend besser zu erkennen, haben wir die Datenpunkte durch polynomielle Regression angenähert.

Anhand der Abbildung können wir folgende Anmerkungen machen: die Länge des Lösungsweges beeinflusst eindeutig sowohl die benötigte Zeit zum Lösen des Spiels als auch die empfundene Schwierigkeit, der Zusammenhang ist in beiden Fällen näherungsweise linear. Auch das ähnliche Maß des durchschnittlichen kürzesten Weges folgt diesem Zusammenhang.

Die Größe des Zustandsraumes, der von dem Spieler exploriert werden muss, scheint weder bei der benötigten Zeit noch bei der Schwierigkeitseinschätzung der Spieler eine Rolle zu spielen. Auch gegenüber der Anzahl der kürzesten Wege sind diese beiden Komplexitätskriterien indifferent. Bemerkenswert ist allerdings, dass die beiden Maße  $avdg$  und  $br$  eine Tendenz zur negativen Korrelation aufweisen. Besonders hohe Werte dieser beiden Maße scheinen die empfundene Komplexität



(a) Maße aufgetragen gegen die Lösungszeit



(b) Maße aufgetragen gegen die empfundene Schwierigkeit

Abbildung 11: Eine Darstellung, inwiefern unsere Maße mit den Versuchsdaten in Einklang gebracht werden können.

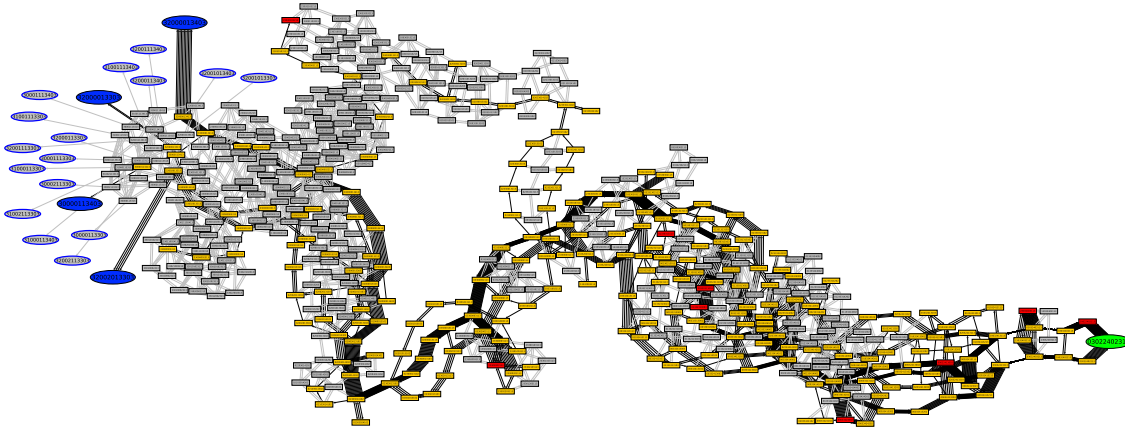


Abbildung 12: Eine Darstellung des Zustandsraumes für das Level 1 19I und aggregierte Informationen, wie die Versuchspersonen sich in diesem bewegt haben: der Startzustand ist grün, besuchte Knoten gelb, nicht besuchte Knoten grau gefärbt. Von Versuchspersonen erreichte Endzustände sind blau gefärbt, nicht besuchte Endzustände sind blau umrandet. Knoten, in denen mindestens eine Versuchsperson das Level abgebrochen hat, sind durch rote Färbung gekennzeichnet. Die Anzahl der schwarzen Kanten zeigen an, wieviele Versuchspersonen diesen Übergang gewählt haben.

eher hinabzusetzen. Erst ab einem gewissen Wert ist die empfundene Komplexität unabhängig von diesen Maßen.

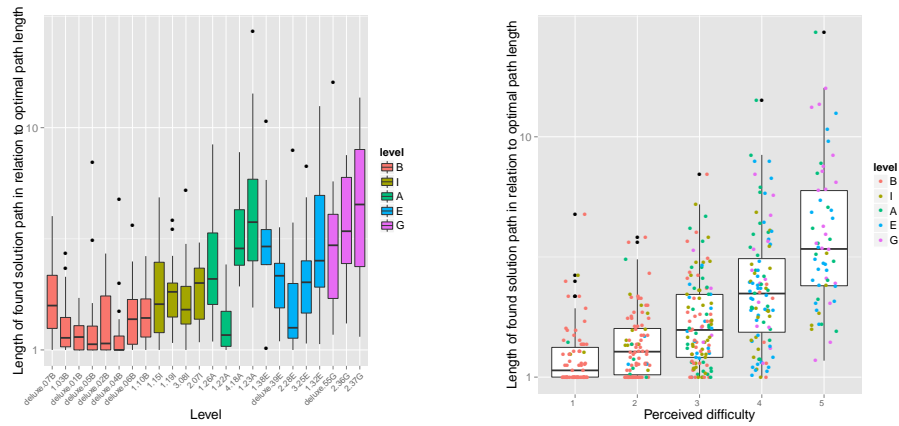
Ebenfalls auffällig ist, dass die Anzahl der Autos auf dem Spielbrett beziehungsweise die Anzahl der belegten Felder einen größeren Beitrag zur Komplexität leisten als bisher angenommen. Vor allem bei der Einschätzung der Spieler ist ein stetiger Trend nach oben zu erkennen.

#### 4.2.5 Irrwege im Zustandsraum

Mehrere Versuchspersonen haben uns die Rückmeldung gegeben, dass sie sich während des Versuchs eine Art *reset*-Möglichkeit gewünscht hätten, sodass es möglich gewesen wäre, ein Level von vorne zu beginnen, wenn man nicht weiterkam. Allerdings haben wir uns im Vorhinein bewusst gegen diese Möglichkeit entschieden, um die Versuchspersonen zu zwingen, sich frei im Zustandsraum zu bewegen, und im Nachhinein prüfen zu können, auf welche Art und Weise die Spieler diese Exploration durchführen. Diese Rückmeldungen geben uns jedoch einen Hinweis darauf, dass vielleicht nicht unbedingt die Größe, aber möglicherweise die Struktur des Zustandsraumes die empfundene Komplexität eines Levels beeinflussen.

Deshalb betrachten wir im Folgenden die Problemräume und die Navigation der Versuchspersonen in ihnen etwas genauer. In Abbildung 12 sieht man beispielhaft den Problemraum des Levels 1 19I. Sowohl in diesem als auch allen anderen ist deutlich zu erkennen, dass Versuchspersonen bevorzugt die gleichen Wege nehmen und oft einen großen Teil des Zustandsraumes abdecken. Wir vermuten, dass Versuchspersonen, die in Teilen des Problemraumes Schwierigkeiten hatten, sich zu orientieren und sich in diesem Teil „verirren“, dieses Level eher als komplex empfinden. Doch wie kann die Eigenschaft, dass eine Versuchsperson sich im





(a) Auf der  $x$ -Achse sind alle von Versuchspersonen gespielte Level, auf der  $y$ -Achse ist das Verhältnis zwischen der Länge des Lösungsweges des Spielers zur Länge des optimalen Lösungsweges dieses Levels. Die Boxplots mitteln über alle Spieler, die das jeweilige Level gespielt haben. Die Färbung gibt einen Hinweis auf die Schwierigkeitsangabe des Herstellers des betreffenden Levels.

(b) Auch hier ist auf der  $y$ -Achse das Verhältnis zwischen der Länge der Lösung eines Spielers zu der optimal möglichen Lösungslänge, auf der  $x$ -Achse hier jedoch die Schwierigkeitseinschätzung der Versuchsperson aufgetragen (1 für *sehr einfach* bis 5 für *sehr schwer*). Jeder Datenpunkt ist also ein gespieltes Spiel eines Spielers, dargestellt sind alle gelösten Level aller Spieler. Aus Darstellungsgründen wurden die Daten zusätzlich in  $x$ -Richtung gestreut (*jittering*).

Abbildung 13: Dargestellt ist der Zusammenhang zwischen Pfadlänge und Level bzw. Schwierigkeitseinschätzung der Spieler. Enthalten sind alle gelösten Spiele aller Versuchspersonen. Man beachte die logarithmische Skala auf der  $y$ -Achse.

Problemraum „verirrt“, überhaupt quantifiziert werden? Als erster naiver Ansatz dient folgende Vermutung: Hat eine Versuchsperson Schwierigkeiten, einen Pfad vom Startzustand zu einem Endzustand zu finden, wird sie mehr Züge benötigen, als notwendig sind. Um dies zu überprüfen, betrachten wir für jeden Spieler und jedes Level, wieviele Züge er benötigt hat, in Relation zur minimal notwendigen Anzahl an Zügen. In Abbildung 13 ist diese Beziehung dargestellt.

Interessant ist, dass dieses Maß sowohl mit der Herstellerangabe zur Komplexität als auch mit der empfundenen Komplexität korreliert. Betrachten wir Abbildung 13a genauer, so können wir erkennen, dass es durchaus einen Zusammenhang gibt zwischen der vom Hersteller angegebenen Schwierigkeit der Level und dieser einfachen Quantifizierung des „Herumirrens“ im Problemraum des Spiels. Wir sehen ebenso, dass diese Darstellung in Einklang steht mit Abbildung 9, da es ab den *intermediate*-Spiele kaum noch Versuchspersonen gibt, die das Level optimal gelöst haben. Bemerkenswert ist allerdings auch, dass nach diesem Kriterium zum einen das Level 1 22A ein auffälliger Ausreißer nach unten ist, zum anderen die *expert*-Spiele nicht in das Schema zu passen scheinen. Weiterhin beobachten wir, dass die Streuung dieses Maßes größer ist, je höher die Schwierigkeitseinschätzung durch den Hersteller ausfällt.

Werfen wir einen Blick auf Abbildung 13b, so fällt uns noch etwas anderes auf: die Anzahl überflüssiger Züge, die Versuchspersonen im Problemraum des Levels machen, korreliert direkt mit ihrer eigenen Schwierigkeitseinschätzung und hier



ist erkennbar, dass die Herstellerangabe der Schwierigkeit in beiden Dimensionen weit streut, die gefühlte Schwierigkeit also eher von der Orientierungslosigkeit abhängt als das von der Herstellerangabe vorhergesagt wird.

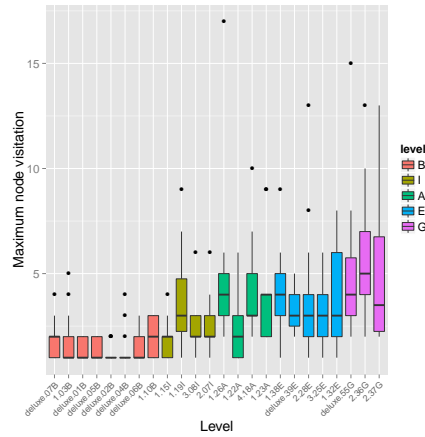
Um die Idee des „Verirrens“ noch ein wenig genauer zu untersuchen, stellen wir folgende Idee vor: Hat eine Versuchsperson Schwierigkeiten, einen Pfad zu finden, um einen Teil des Netzwerkes zu verlassen und in einen anderen – einem Endzustand näheren – Netzwerkteil zu gelangen, wird sie vermutlich einige Knoten mehrfach besuchen. Je schwerer ihr das Finden des Weges fällt, desto häufiger wird sie einige Knoten besuchen nur um festzustellen, dass sie an diesen Knoten doch nicht weiterkommt. Offensichtlich deckt dieser Ansatz nicht alle Fälle ab, da Versuchspersonen vermutlich mehrere Möglichkeiten ausprobieren, um einen Pfad in die richtige Richtung zu finden, jedoch kann die Tatsache, dass eine Versuchsperson einen Knoten mehrfach besucht hat, ein klares Anzeichen dafür sein, dass sie Schwierigkeiten hat, einen Lösungsweg zu finden.

Auf dieser Grundidee basierend definieren wir für eine Versuchsperson die *visitation* eines Knotens in dem Level als die Anzahl der Besuche dieses Knotens im Verlauf des Spiels der Versuchsperson. Im Folgenden werden wir die *maximum node visitation* und die *average node visitation* eines Spiels einer Versuchsperson betrachten: ersteres ist der maximale Wert aller *node visitations* dieser Versuchsperson in diesem Spiel, letzteres der Durchschnittswert über alle besuchten Knoten dieser Versuchsperson in diesem Spiel. Sofort fällt auf, dass die Abbildungen 14a und 14c beziehungsweise die Abbildungen 14b und 14d sich bis auf die Skalen sehr ähneln und große Parallelen zu Abbildung 13 aufweisen, was durch die Verwandtschaft der zugrunde liegenden Ideen begründet ist. Auch dieses Maß der „Verirrung“ scheint die empfundene Komplexität eines Spiels gut abzubilden: es ist die klare Tendenz zu erkennen, dass sowohl die *maximum* als auch die *average node visitation* die Schwierigkeitseinschätzung der Spieler widerspiegelt.

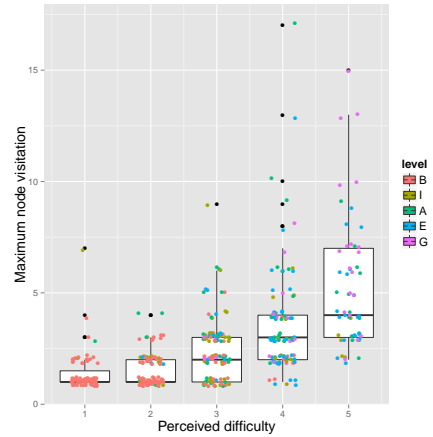
Es bleiben jedoch noch einige Fragen offen. Zum einen ist genauer zu untersuchen und zu evaluieren, wie quantifiziert werden kann, dass Versuchspersonen sich im Zustandsraum „verirren“. Zum anderen ist nicht klar, ob sich die Schwierigkeiten von Versuchspersonen, sich im Zustandsraum zurechtzufinden, allein auf dessen strukturellen Eigenschaften reduzieren lassen. Sollte dies der Fall sein, wäre es aber interessant, die Netzwerke auf diese Eigenschaften hin zu untersuchen. Vorstellbar wären beispielsweise Analysen, die die Zustandsräume auf sogenannte *bridges* überprüfen, also Kanten, durch deren Entfernen der Graph in zwei Zusammenhangskomponenten zerfällt. Wenn *bridges* in allen möglichen Lösungswegen enthalten sind, könnte ihre Existenz zur Komplexität beitragen.

#### 4.2.6 Analyse der Zustände

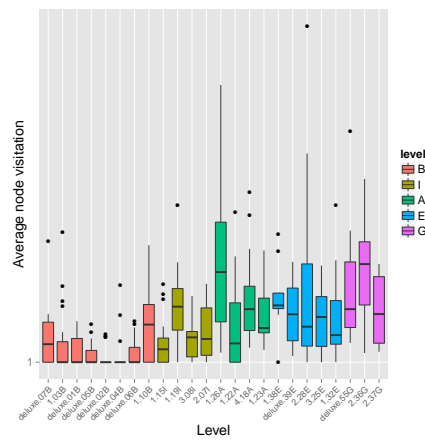
In den vorherigen Betrachtungen haben wir vor allem globale Eigenschaften der Spiele und Versuchsdaten in den Mittelpunkt gestellt. Im Folgenden vertiefen wir die Analyse der einzelnen Konfigurationen des Spiels weiter. Wir fragen uns, ob es bestimmte Spielzustände gibt, in denen die Spieler besonders lange überlegt, in denen sie abgebrochen oder in denen sie eventuell sogar Fehler gemacht haben. Gibt es Eigenschaften, die diese *Zustände* kennzeichnen und von den übrigen unterscheiden?



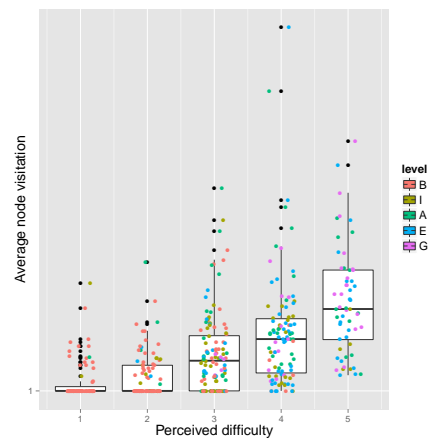
(a) *Maximum node visitation* gegen die einzelnen Level aufgetragen. Die Boxplots mitteln über alle *maximum node visitations* aller Spieler, die dieses Level gespielt haben.



(b) *Maximum node visitation* gegen die Schwierigkeitseinschätzung der Spieler aufgetragen. Die Daten wurden in *x*-Richtung gestreut (*jittering*), um eine bessere Darstellung zu ermöglichen.



(c) *Average node visitation* gegen die Level aufgetragen, gemittelt wird auch hier über die Werte aller Spieler, die dieses Level gespielt haben. Man beachte die logarithmische Skala auf der *y*-Achse.



(d) *Average node visitation* gegen die Schwierigkeitseinschätzung der Spieler aufgetragen. Auch hier wurden die Datenpunkte in *x*-Richtung gestreut (*jittering*). Man beachte die logarithmische Skala auf der *y*-Achse.

Abbildung 14: Eine Übersicht der *node visitations* in allen Spielen.

Um eine bessere Vorstellung davon zu erhalten, in welcher Weise die Versuchspersonen sich durch den Problemraum bewegt haben und in welchen Zuständen sie gezögert haben, betrachte man Abbildung 15: Sichtbar sind alle vom Startzustand aus erreichbaren Spielzustände. Von Spielern besuchte Knoten sind dabei farbig, nicht besuchte Knoten sind grau. Der Startzustand ist schwarz umrandet und elliptisch, Endzustände sind blau gefärbt (wenn sie von Spielern besucht wurden) oder violett umrandet. Zustände, in denen Spieler das Level abgebrochen haben, verfügen über eine rote Umrandung. Die Anzahl der Kanten zeigt, wieviele Versuchspersonen einen gegebenen Übergang gewählt haben. Kanten sind rot bzw. rosa gefärbt, wenn es sich bei diesem Zug um einen unintuitiven Zug (siehe Definition in Kapitel 2) handelt.

Diese Abbildung enthält in den Knotenfärbungen kodiert die Zeit, die Spieler durchschnittlich dort verbracht haben, bevor sie ihn durch einen Zug verlassen haben. Auch hier werden für die Durchschnittsbildung Spielerzeiten, die 10 Minuten überschreiten, als Pausen aufgefasst und gehen nur mit 10 Minuten in den Durchschnitt ein. Diese Abbildung vermittelt eine qualitative Vorstellung.

Zunächst betrachten wir die Überlegungszeiten etwas genauer. Ersichtlich ist, dass der Großteil der Knoten von den Spielern innerhalb weniger Sekunden wieder verlassen wird. Ein Histogramm verdeutlicht die Verteilung der Überlegungszeiten über aller Konfigurationen und Spiele (siehe Abbildung 16a). Wir wollen uns jedoch die Zustände anschauen, in denen die Versuchspersonen länger überlegen. Dazu greifen wir noch einmal auf unsere vorgeschlagenen Komplexitätsmaße aus Kapitel 3 zurück. Einige dieser Maße waren knotenweise definiert und können demnach auch auf einzelne Knoten angewendet werden: dies sind der Knotengrad *degree*, die *branching complexity* *br* und die Anzahl bewegbarer Autos *mc*. Uns interessiert, ob ein Zusammenhang gefunden werden kann zwischen den Werten dieser Maße und der Zeit, die Spieler an diesem Knoten überlegt haben. Diese potentiellen Zusammenhänge untersuchen wir einzeln. Jedoch haben wir in vorherigen Analysen ja beobachten können, dass nur ein geringer Teil der Spiele optimal gelöst wird. Da die *branching complexity* aber nur auf Knoten des optimalen Weges Sinn ergibt, betrachten wir für dieses Maß zusätzlich nur die *beginner*-Spiele einzeln, da lediglich in diesen noch ausreichend Spieler optimale Wege gefunden haben.

In Abbildung 16b ist eine Visualisierung dieser Analyse zu sehen: Für jeden besuchten Knoten in jedem Spiel ist dort die durchschnittliche Überlegungszeit aller Spieler gegen den Wert eines der drei Maße aufgetragen. Eine direkte Korrelation zwischen den Maßen und der Überlegungszeit gibt es allerdings nicht, weder eine positive noch eine negative.

Betrachtet man die beiden Maße *degree* und *mc* genauer, erkennt man jedoch die Tendenz, dass die Überlegungszeit geringer ist, wenn diese beiden Maße besonders niedrig oder besonders hoch sind, die Überlegungszeit ist in der Mitte maximal. Wenn wir uns vor Augen führen, was diese Maße aussagen, dann ist dieses Ergebnis plausibel: Wenn der Grad beziehungsweise die Anzahl der bewegbaren Autos niedrig ist, hat die Versuchsperson nur wenig Auswahl bei seinem Zug, was sie bis zu einem gewissen Punkt auch schnell erfassen und umsetzen kann. Werden die Auswahlmöglichkeiten jedoch mehr, ist dies nicht mehr „mit einem Blick“ zu erfassen und der Spieler zögert länger. Erhöht sich die Anzahl

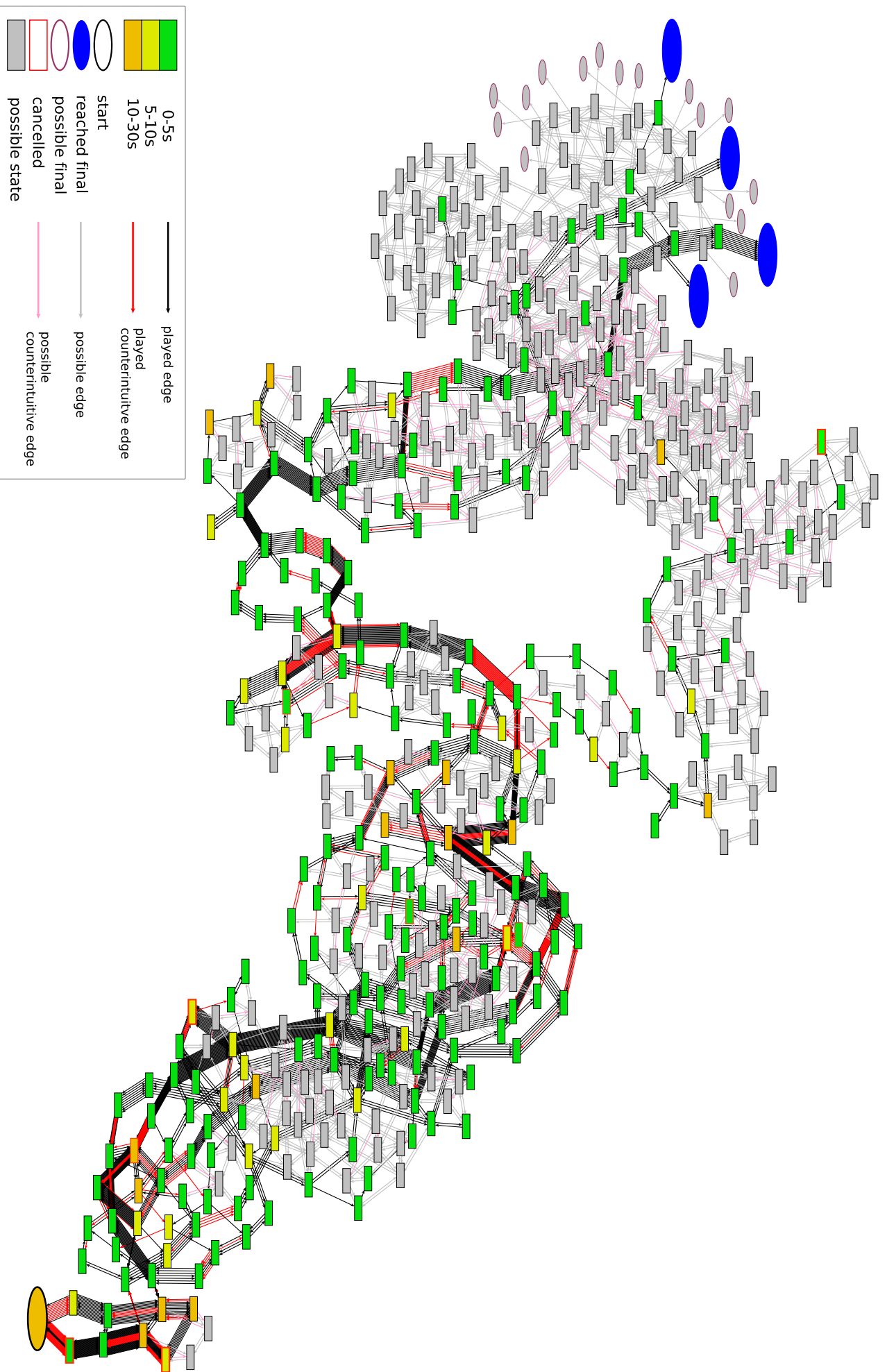
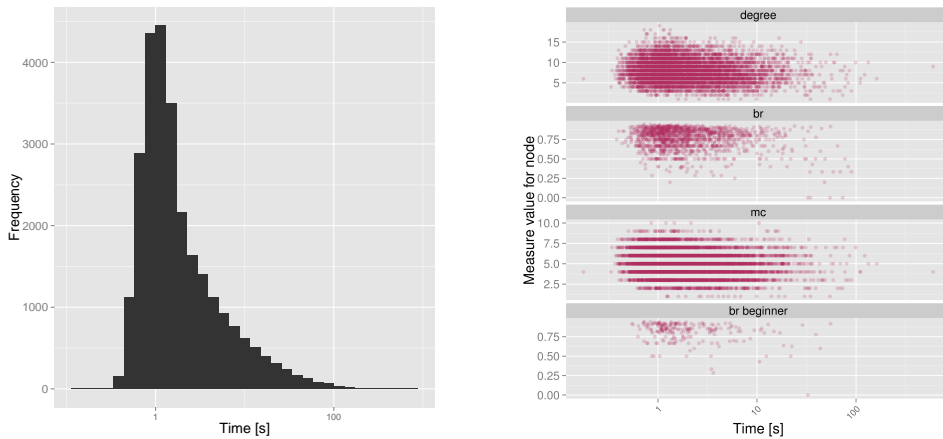


Abbildung 15: Der gleiche Zustandsraum wie in Abbildung 12, allerdings mit zusätzlichen Informationen.



(a) Die Verteilung der Überlegungszeiten der Spieler. Man beachte die logarithmische Skala auf der x-Achse.

(b) Dargestellt sind alle besuchten Spielzustände aller Spieler und aller Level: Auf der x-Achse ist die Überlegungszeit der Spieler an diesem Knoten in Sekunden, auf der y-Achse der Wert des jeweiligen Maßes an diesem Knoten. Ein Datenpunkt aggregiert in einem Durchschnittswert die Zeiten aller Spieler, die diesen Knoten besucht haben. Man beachte die logarithmische auf der x-Achse.

Abbildung 16: Eine Analyse der Überlegungszeiten der Spieler

an Auswahlmöglichkeiten noch weiter, ist die Anzahl an blockierenden Autos vermutlich so gering, dass sich dem Spieler keinerlei Hindernisse bieten, wegen derer er zögern würde.

Die Betrachtung des Maßes *br* bietet leider keine signifikanten Ergebnisse. Auch wenn man nur die *beginner*-Spiele betrachtet, kann aufgrund dieser Daten keine Aussage getroffen werden. In weiteren Analysen könnte man aber beispielsweise die Definition des Maßes erweitern, damit es nicht nur auf dem optimalen Graphen, sondern auf dem gesamte Problemraum definiert ist. Eine naive Idee für die erweiterte *branching complexity* für einen Knoten wäre hier der Anteil der ausgehenden Kanten, die den restlichen Pfad zu einem Zielzustand nicht verlängern, in Relation zu allen ausgehenden Kanten dieses Knotens. Für jeden Knoten ist der Abstand  $d$  zum nächsten Zielzustand bekannt, die Kanten, die den restlichen Pfad nicht verlängern, wären also jene Kanten, deren inzidenter Knoten einen Abstand zum Zielzustand  $d' < d$  besitzt. Es wäre interessant, die vorliegenden Daten mit diesem Ansatz nochmals zu untersuchen.

Weiterhin haben wir uns gefragt, in welchen Spielzuständen die Spieler dazu tendieren, Fehler zu machen. In einem ersten Ansatz betrachten wir die Knoten, die auf einem optimalen Pfad zu einem Spielzustand liegen, in denen Spieler sich jedoch für einen Zug entschieden haben, mit dem sie den optimalen Pfad verlassen. Leider hat dieser Ansatz zu keinen Erkenntnissen geführt und wird daher an dieser Stelle nicht weiter vertieft. Wir können daher keine Aussage darüber machen, ob es einen Zusammenhang zwischen unseren Knotenmaßen und der Tendenz der Spieler gibt, einen falschen Zug zu machen.

4.2.7 *Unintuitive Züge*

In Kapitel 3 haben wir das Konzept der unintuitiven Züge eingeführt, auf welches wir unsere Versuchsdaten nun untersuchen. In Abbildung 15 sind diese Züge bereits durch ihre rote beziehungsweise rosa Färbung gekennzeichnet. Unsere Hypothese lautet: Versuchspersonen vermeiden unintuitive Züge, wo es möglich ist und nehmen dafür eventuell sogar längere Lösungswege in Kauf. Zudem vermuten wir, dass Versuchspersonen tendenziell länger zögern, bevor sie einen unintuitiven Zug ausführen.

Die Überprüfung dieser Thesen ist jedoch nicht ohne Weiteres möglich. Zum einen werden Spieler mindestens ein oder zwei Züge vorausplanen, das heißt, die Überlegungszeit, die den unintuitiven Zug betrifft, wird nicht unbedingt unmittelbar vor diesem Zug stattfinden. Zum anderen werden Spieler eventuell voraussehen, dass ein solcher Zug das Resultat einer bestimmten Zugfolge ist und diese dann so nicht ausführen, sondern einen Weg bevorzugen, der diesen Zug später oder gar nicht enthält. Weiterhin heißt die Tatsache, dass ein unintuitiver Zug möglich wäre, noch längst nicht, dass dieser Zug auch sinnvoll ist. Zum Beispiel hat jeder Knoten, in dem der Spieler das rote Auto gerade vorwärts bewegt hat oder ein Auto, das das rote Auto blockiert hat, aus dem Weg geschoben hat (was zwangsläufig während eines Lösungsweges vorkommen muss), eine unintuitive Kante, nämlich diesen Zug rückgängig zu machen. Diese drei Überlegungen legen eine aufwändigere Analyse bezüglich der unintuitiven Züge nahe, die den Rahmen dieser Arbeit sprengen würde.

Wir möchten dennoch einige der einfacheren Ansätze im Folgenden vorstellen. Die oben genannte Überlegungen sollten jedoch bei der folgenden Betrachtung im Hinterkopf behalten werden.

Zunächst möchten wir erneut die Überlegungszeit der Spieler untersuchen und mit der Information, ob es sich beim direkt folgenden Zug um einen unintuitiven Zug handelt, in Beziehung setzen. Hierbei weisen wir noch einmal darauf hin, dass dies kein perfektes Modell eines realen Spielers ist, hoffen aber, dass wir durch diese Analyse das eventuelle Zögern direkt vor einem unintuitiven Zug erfassen können.

In Tabelle 6 haben wir die Züge aller Spieler in allen Leveln untersucht und die Überlegungszeit vor diesen Zügen betrachtet. Man sieht, dass die durchschnittliche Überlegungszeit vor unintuitiven Zügen um 2 Sekunden länger ist, aufgrund der Standardabweichung von 9 bzw. 12 Sekunden impliziert dies jedoch noch nicht

	MEAN	MEDIAN	SD	MAX	MIN	NUMBER
intuitive	3.749 s	1.311 s	9.504 s	197.2 s	0.328 s	22632
counterint.	5.874 s	1.707 s	12.46 s	176 s	0.218 s	4427

Tabelle 6: Mittelwert, Median, Standardabweichung, Maximalwert und Minimalwert der Überlegungszeiten der Spieler in Sekunden, gruppiert nach der Information, ob nach der Überlegungszeit ein intuitiver oder ein unintuitiver Zug folgt. In der letzten Spalte ist die absolute Anzahl der Züge angegeben. Hier wurden vorher alle Züge entfernt, bei denen die Spieler länger als 200 Sekunden überlegt haben.



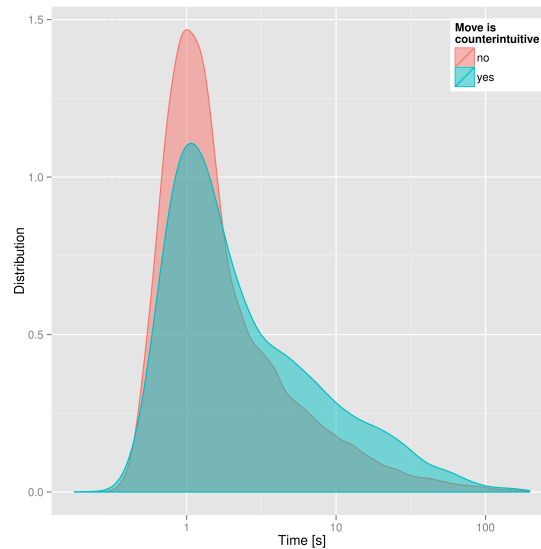


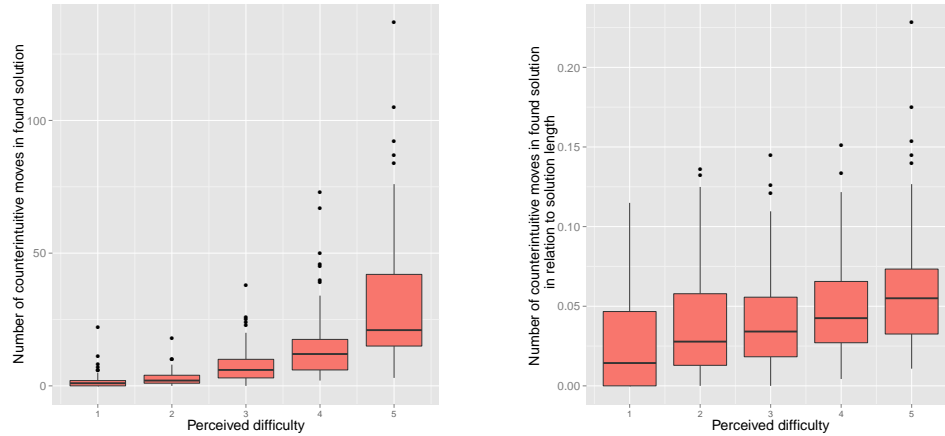
Abbildung 17: Die Verteilung der Überlegungszeiten der Spieler, gruppiert nach dem Kriterium, ob der folgende Zug intuitiv oder unintuitiv ist. 37 Datenpunkte mit einem Wert von über 200 Sekunden wurden vorher entfernt. Man beachte die logarithmische Skala auf der x-Achse.

die Signifikanz des Ergebnisses. Die Beobachtung, dass die Wertebereiche der beiden Gruppen fast identisch sind (wir haben vor der Analyse alle Züge mit einer Überlegungszeit von mehr als 200 Sekunden entfernt), wirft die Frage auf, wie die Verteilung der Überlegungszeiten bei unintuitiven und intuitiven Zügen aussieht, sodass sich die Durchschnittswerte doch um gut 2 Sekunden unterscheiden.

Um diese Frage zu beantworten, kann Abbildung 17 herangezogen werden. Hier ist die Verteilung der Überlegungszeiten der Spieler aufgetragen, das Diagramm basiert auf insgesamt 27059 Datenpunkten, 22632 gehören der Gruppe der intuitiven, 4427 der Gruppe der unintuitiven Züge an. Deutlich zu sehen ist, dass beide Gruppen einen gemeinsamen Peak besitzen, der bei wenigen Sekunden liegt, das heißt, der Großteil der intuitiven sowie ein Großteil der unintuitiven Züge wird sehr schnell entschieden. Der Peak der unintuitiven Züge ist allerdings etwas niedriger. Interessant ist jedoch das Verhalten hinter diesem Peak, die Kurve der intuitiven Züge flacht schneller wieder ab als die Kurve der unintuitiven Züge, um bei etwa 100 Sekunden wieder zusammenzulaufen. Dies ist ein Indiz dafür, dass Spieler tendenziell direkt vor unintuitiven Zügen etwas länger zögern als vor intuitiven. Da dieser Plot knapp 30000 Datenpunkte enthält, können die Unterschiede in den Verteilungen als deutlicher Hinweis in diese Richtung interpretiert werden.

Allerdings wird in dieser Darstellung nicht erfasst, dass Spieler eventuell zögern, um dann einen unintuitiven Zug gerade *nicht* zu machen und die längere Überlegungszeit dann darauf zurückzuführen ist, dass sie einen anderen Pfad nehmen. Ebenfalls wird die Vermutung, dass Spieler einige Züge vorausplanen hier nicht adressiert. Wenn sich diese beiden Vermutungen bestätigen ließen, würde die Tendenz vermutlich noch stärker ausfallen.

Wir betrachten nun die Spiele, in denen die Spieler überhaupt unintuitive Züge gemacht haben. Hierunter fallen auch eventuelle Rückwärtskanten, wenn sie einen Zug, in dem sie das rote Auto vorwärts bewegen oder ein das rote Auto blockieren-



(a) Der Zusammenhang zwischen empfundener Schwierigkeit und absoluter Anzahl an unintuitiven Zügen im Lösungsweg. Die Boxplots mitteln über alle Lösungswege aller Spieler in allen Spielen mit der jeweiligen Schwierigkeitseinschätzung (1 für *sehr einfach* bis 5 für *sehr schwer*).

(b) Der Zusammenhang zwischen empfundener Schwierigkeit und dem Verhältnis zwischen Anzahl an unintuitiven Zügen im Lösungsweg und Länge des Lösungsweges. Die Boxplots mitteln auch hier über alle Lösungswege aller Spieler aller Spiele mit der jeweiligen Schwierigkeitseinschätzung.

Abbildung 18: Zusammenhang zwischen empfundener Schwierigkeit und unintuitiven Zügen.

des Auto aus dem Weg schieben, rückgängig machen, da dies auch unintuitive Züge in unserem Sinne sind. Ob die unintuitiven Züge, die die Spieler gemacht haben, in dem Spielverlauf überhaupt sinnvoll waren, bleibt hierbei unbeachtet, wir betrachten nur die absolute Anzahl der gemachten unintuitiven Züge. Wir vermuten, dass Spiele, in denen Spieler mehr unintuitive Züge gemacht haben (nicht-machen mussten), von ihnen als schwieriger empfunden werden. Dazu betrachten wir für jeden Spieler und für jedes Level, das er gespielt hat, die absolute Anzahl an unintuitiven Zügen in seinem Lösungsweg und setzen dies in Beziehung zu der Schwierigkeitseinschätzung, die der Spieler zu diesem Spiel abgegeben hat. In [Abbildung 18a](#) ist deutlich zu sehen, dass diese beiden Werte stark korrelieren und eventuell sogar ein quadratischer Zusammenhang besteht. Da der Zusammenhang zwischen empfundener Schwierigkeit und der Länge des Lösungsweges bereits gezeigt wurde und in einem längeren Lösungsweg vermutlich mehr unintuitive Züge enthalten sind, ist in [Abbildung 18b](#) eine entsprechende Modifizierung zu sehen, in der die Anzahl der unintuitiven Zügen in einem Lösungsweg mit der Länge des Lösungsweges normiert wurde, also mit der Gesamtanzahl der Züge, die dieser Spieler in diesem Spiel benötigt hat. Auch wenn der Zusammenhang damit schwächer wird, ist hier immer noch ein Trend zu erkennen, dass mit steigender Anzahl unintuitiver Züge das Spiel als schwieriger empfunden wird. Wir sehen also Hinweise dafür, dass sich die beiden Hypothesen, dass Spieler ein Spiel mit vielen unintuitiven Zügen als komplexer wahrnehmen und dass Spieler direkt vor unintuitiven Zügen länger zögern als vor intuitiven bestätigen können. Dennoch bleiben einige Fragen offen, die weitere Analysen verlangen.

Die Vermutung, dass Versuchspersonen unintuitive Züge meiden und deshalb eventuell sogar längere Lösungswege in Kauf nehmen, lässt sich auch unter dem



Kompromiss, leicht verfälschte Werte zu erhalten, nicht simpel analysieren. Um allerdings eine Idee davon zu bekommen, ob sich für diese Hypothese überhaupt eine weitere Untersuchung lohnt, möchten wir einen kleinen Ausschnitt aus einem Zustandsraum betrachten und beispielhaft untersuchen, welche Züge die Versuchspersonen bevorzugt gewählt haben, ohne den Anspruch auf Verallgemeinerung zu erheben.

Dazu betrachten wir das Level 1 10B in der in Abbildung 15 eingeführten Notation. Abbildung 19 zeigt einen Ausschnitt dieses Zustandsraumes, der einen ausgezeichneten Knoten mit dem Label 22440034114 besitzt (in der Abbildung als *special node* bezeichnet), der in allen möglichen Lösungswegen enthalten sein muss. Ist dieser Knoten erreicht, ist der weitere Lösungsweg im Wesentlichen eine Kette von eindeutigen Zügen, die innerhalb weniger Züge zu einem der beiden Lösungszustände führt und daher nicht abgebildet ist. In keinem der Knoten hinter diesem ausgezeichneten Knoten hat der Spieler mehr die Möglichkeit „folgeschwere“ Fehler zu machen. Auch ist in dem in Abbildung 19 weggelassenen Teilgraphen kein einziger unintuitiver Zug mehr notwendig, um einen Lösungszustand zu erreichen. Nur in drei Knoten wären unintuitive Züge als Rückwärtskanten möglich.

Aus diesen Gründen betrachten wir nur den gezeigten Ausschnitt des Zustandsraumes und möchten beispielhaft in einigen Knoten die Entscheidungen der Spieler bezüglich ihres nächsten Zuges nachvollziehen. Dazu berücksichtigen wir einerseits die Möglichkeiten, die die Spieler in diesen Knoten haben, für welchen Zug sich die Spieler entscheiden und ob die möglichen Züge unintuitiv sind. Zudem können wir uns ansehen, ob die möglichen Züge einen Umweg darstellen: angenommen, ein Spieler befindet sich an Knoten  $v$  und der minimale Abstand des Knotens  $v$  zu einem Endzustand beträgt  $n$  Züge. Die Kante  $(v, w)$  stellt einen Umweg dar, falls der minimale Abstand von  $w$  zu einem Endzustand mehr als  $n - 1$  Züge beträgt.

In Tabelle 7 ist für alle Knoten des Teilgraphen aufgelistet, welche Nachfolgeknoten er hat, von wievielen Versuchspersonen diese Kante gewählt wurde und ob diese Kante unintuitiv ist sowie ob sie einen Umweg, also *indirect* ist, darstellt.

Beginnen wir im Startzustand 32441033414, der auch in Abbildung 20a dargestellt ist. Man sieht, dass die Spieler genau zwei Zugmöglichkeiten haben: das waagerechte schwarze Auto der Länge 3 oder das rote Auto um ein Feld rückwärts bewegen. Ersteres wäre ein intuitiver, zweiteres ein unintuitiver Zug, für ersteres entscheiden sich 28 Spieler, für zweiteres nur 6 Spieler. Jedoch ist dies in diesem Fall nicht überraschend, beide Möglichkeiten liegen auf dem optimalen Weg und der erstere Zug ist insofern naheliegender als dass er das rosa Auto befreit, während der unintuitive Zug erstmal keine Blockaden auflöst. Die Aktion, das waagerechte schwarze Auto rückwärts zu bewegen, müssen die Spieler, die das rote Auto bewegt haben, direkt nach diesem Zug auch durchführen. Zwei Spieler machen den Zug, das rote Auto zu bewegen, allerdings sofort wieder rückgängig.

Betrachten wir Zustand 32440034414 (Abbildung 20b): es gibt neben der Rückwärtskante drei Zugmöglichkeiten, das rote Auto ein Feld rückwärts, das hellblaue Auto zwei Felder nach links oder das rosa Auto ein Feld nach oben zu bewegen.

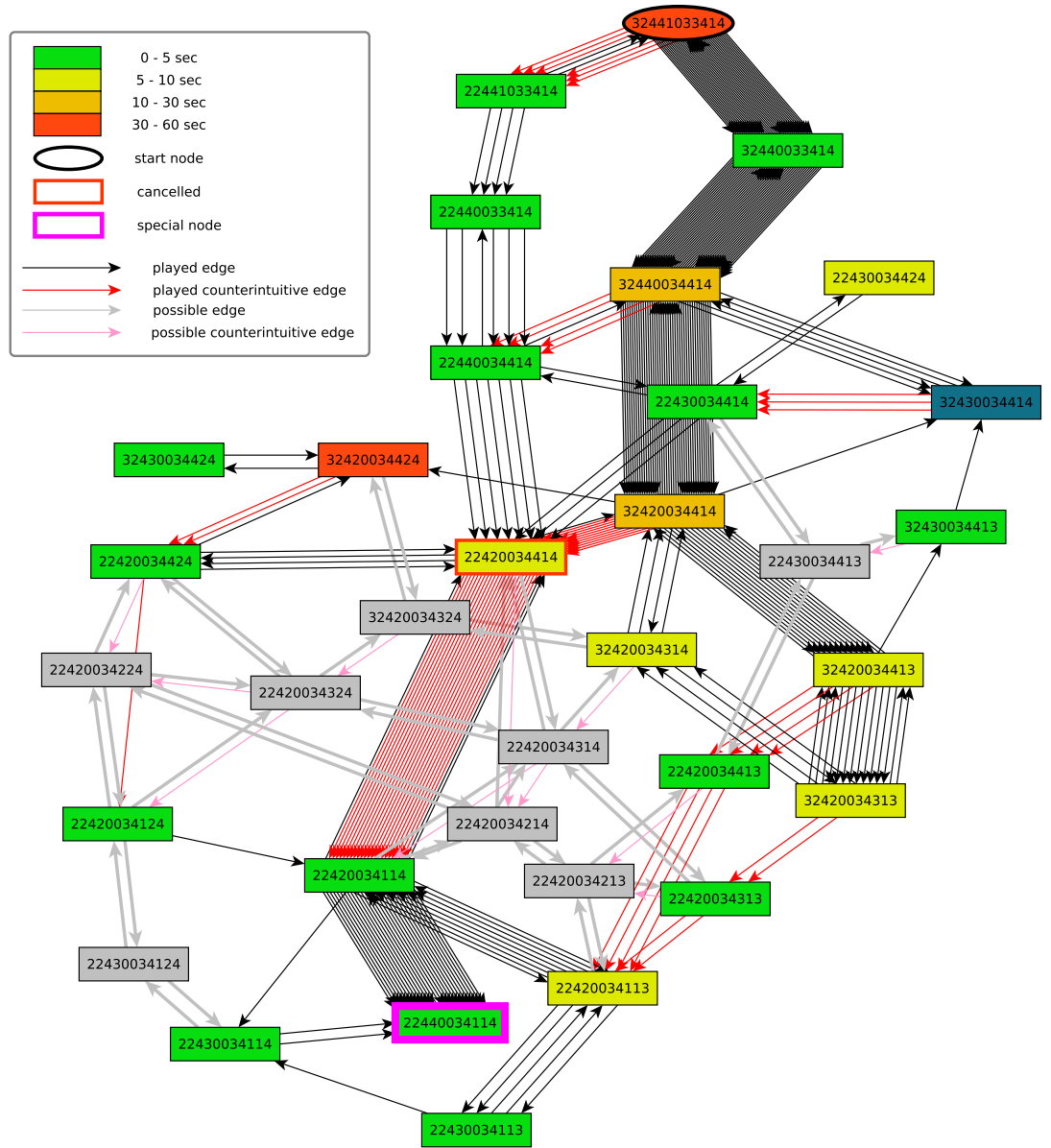


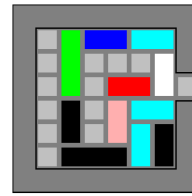
Abbildung 19: Ein Ausschnitt aus dem Zustandsraum zu Level 1 10B.

Die ersten beiden stellen keinen Umweg dar, jedoch ist das erste ein unintuitiver Zug nach unserer Definition. Die beiden anderen Möglichkeiten sind Umwege. Interessanterweise entscheiden sich 29 dafür, das hellblaue Auto zu bewegen, nur 4 Spieler bewegen das rote Auto rückwärts, immerhin insgesamt 13 Spieler wählen die Kanten, die in einen Umweg resultieren.

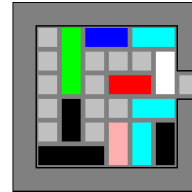
Besonders eindeutig wird es, wenn wir den Knoten betrachten, den wir erhalten, wenn das hellblaue Auto bewegt wird, also Zustand 32420034414, siehe Abbildung 20c. Den Spielern bieten sich sechs Möglichkeiten: das rote Auto um ein Feld nach links, das horizontale hellblaue Auto um ein oder um zwei Felder nach rechts, das weiße Auto um ein Feld nach unten, das senkrechte hellblaue oder das senkrechte schwarze Auto um ein Feld nach oben bewegen. Hier ist nur der unintuitive Zug, das rote Auto um ein Feld nach links zu bewegen, *kein* Umweg. Alle anderen Möglichkeiten sind zwar nicht unintuitiv, aber Umwege. Trotzdem entscheiden sich nur 10 Spieler für diesen Zug, 26 weitere entscheiden sich für die anderen fünf Möglichkeiten, davon sogar 11 für die Rückwärtskante. 12 entscheiden sich dafür das schwarze Auto um ein Feld nach oben zu bewegen.

Betrachten wir als letztes Beispiel den Knoten, in den man gelangt, wenn man sich im Zustand in Abbildung 20c für die letztgenannte Möglichkeit entscheidet, nämlich Zustand 32420034413, siehe Abbildung 20d. Auch hier steht der unintuitive Zug, das rote Auto um ein Feld nach links zu bewegen, zur Auswahl, was in diesem Fall wirklich der einzige wäre, der keinen Umweg darstellt. Alle anderen Züge, das horizontale hellblaue Auto um ein Feld nach rechts, das senkrechte hellblaue Auto um ein Feld nach oben zu bewegen oder den vorherigen Zug rückgängig zu machen, indem das schwarze senkrechte Auto wieder um ein Feld nach unten geschoben wird, verlängern den möglichen Weg von diesem Knoten zu einem Lösungszustand. Trotzdem entscheiden sich nur 4 Spieler für diese Möglichkeit, 12 Spieler dagegen für die anderen drei Möglichkeiten.

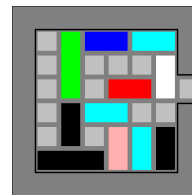
Noch in einigen anderen Knoten dieses Teilgraphen lassen sich ähnliche Beobachtungen machen, und auch wenn die obigen Ausführungen auf einem qualitativen Niveau sind, ist eine deutliche Neigung der Spieler zu erkennen, sich so lange wie möglich vor unintuitiven Zügen zu „drücken“. Bevor hier Schlüsse gezogen werden können und diese Hypothese bestätigt oder vielleicht doch widerlegt werden kann, müssen weitere Analysen systematisch und auf Basis größerer Datenmengen durchgeführt werden.



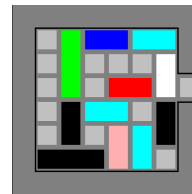
(a) Startzustand  
32441033414



(b) Zustand  
32440034414



(c) Zustand  
32420034414



(d) Zustand  
32420034413

Abbildung 20:  
Ausgewählte  
Zustände aus  
dem Zustandsraum  
von Level 10B.

NODE	SUCCESSOR	PLAYER	COUNTERINTUITIVE	INDIRECT
32441033414	22441033414	6	1	0
	32440033414	28	0	0
32440033414	32441033414	6	0	1
	32440034414	32	0	0
32440034414	32440033414	10	0	1
	22440034414	4	1	0
	32430034414	3	0	1
	32420034414	29	0	0
22440034414	32440034414	1	0	1
	22440033414	1	0	1
	22420034414	7	0	0
	22430034414	1	0	1
32420034414	32440034414	11	0	1
	22420034414	10	1	0
	32420034413	12	0	1
	32430034414	1	0	1
	32420034314	1	0	1
	32420034424	1	0	1
22420034414	32420034414	1	0	1
	22420034424	1	0	1
	22420034114	20	1	0
32420034314	32420034414	3	0	1
	32420034313	1	0	1
32420034424	32430034424	1	0	1
	22420034424	2	1	0
32430034424	32420034424	1	0	0
22420034424	32420034424	1	0	0
22420034424	32420034414	1	0	1
22420034124	22420034114	1	0	0

NODE	SUCCESSOR	PLAYER	COUNTERINTUITIVE	INDIRECT
22420034114	22430034114	1	0	1
	22440034114	25	0	0
	22420034113	2	0	1
	22420034414	1	0	1
22430034114	22440034114	2	0	0
22430034113	22430034114	1	0	0
	22420034113	2	0	1
22441033414	32441033414	2	0	1
	22440033414	4	0	0
22440033414	22440034414	5	0	0
32430034414	32440034414	2	0	1
	22430034414	3	1	0
32430034413	32430034414	1	0	1
22430034414	22430034424	1	0	1
	22440034414	1	0	1
	22420034414	3	0	0
22430034424	22430034414	1	0	1
32420034413	32430034413	1	0	1
	22420034413	1	1	0
	32420034313	9	0	1
	32420034414	3	0	1
32420034313	32420034413	5	0	1
	22420034313	2	1	0
	32420034314	3	0	1
22420034413	22420034113	4	1	0
22420034313	22420034113	2	1	0
22420034113	22420034114	7	0	0
	22430034113	3	0	1

Tabelle 7: Nähere Betrachtung der Spielerentscheidungen zu Abbildung 19

# 5

---

## ZUSAMMENFASSUNG UND AUSBLICK

---

### 5.1 ZUSAMMENFASSUNG DER ERGEBNISSE

Wir haben im Rahmen dieser Arbeit viele überraschende Erkenntnisse gewinnen können. Allerdings sind einige erwartete Hypothesen nicht bestätigt worden. Wir haben für das Spiel *Rush Hour* strukturelle und auf den Eigenschaften des Problemraums basierende Komplexitätsmaße entwickelt und diese auf die vom Hersteller *Thinkfun* vorgeschlagenen Spielkonfigurationen angewendet. Neben den Schwierigkeitseinschätzungen des Herstellers haben wir als Vergleich Daten aus unserem Experiment mit 97 Datensätzen herangezogen.

Zunächst konnten wir feststellen, dass allein die Werte der einzelnen Maße zum großen Teil nicht eindeutig mit der Schwierigkeitsangabe des Herstellers korrelieren. Offensichtlich ist hauptsächlich die Länge der Lösung das Kriterium, nach dem der Hersteller die Startkonfigurationen nach Schwierigkeit klassifiziert. Interessanterweise zeigen einige Maße wie zum Beispiel die *branching complexity* sogar Tendenzen zu negativen Korrelationen mit der Schwierigkeitsangabe und stehen damit im Widerspruch zu unseren Hypothesen.

Die Untersuchung der Maße auf Korrelationen untereinander hat keine überraschenden Erkenntnisse gebracht, wir konnten sehen, dass die meisten Maße untereinander keinerlei Korrelation aufweisen. Der Ansatz, die untersuchten Startkonfigurationen aufgrund ihrer Komplexitätsmaße mithilfe algorithmischer Methoden zu klassifizieren, könnte mit verfeinerteren Methoden lohnend sein. Wir konnten jedoch zeigen, dass ein *k-means-clustering* für diese Daten nicht geeignet ist.

Durch unsere durchgeführte Studie konnten wir weitere interessante Erkenntnisse gewinnen. Wir konnten zeigen, dass zwar nicht die Größe, aber doch die Struktur des Problemraums die empfundene Schwierigkeit eines Problems beeinflusst. Man kann eventuell Parallelen zu Navigationsproblemen ziehen, indem man die Orientierung der Versuchspersonen im Zustandsraum betrachtet.

Zudem haben wir die Tendenz beobachtet, dass ein niedriger Knotengrad in den Problemzuständen die Komplexität eher steigert als senkt. Dies ist insofern eine interessante Beobachtung, als dass – nach unserer Hypothese – die Zugauswahl aus einer größeren Menge schwieriger sein sollte. Unseren Ergebnissen zufolge ist dies jedoch nicht der Fall.

Beispielhaft konnten wir zeigen, dass im Spiel *Rush Hour* Züge existieren, die von Versuchspersonen bei der Lösung möglichst vermieden werden, obwohl sie zur Lösung notwendig wären. Dies deckt sich mit der in der Psychologie *hill climbing* genannten Heuristik, die Lösungsschritte ausschließt, die scheinbar die

Ähnlichkeit zum Zielzustand verringern. Unsere Versuchspersonen zeigten weiterhin die Tendenz, Züge solcher Art zu vermeiden und benötigten eine längere Zeit, bevor sie einen solchen Zug ausführten.

## 5.2 AUSBLICK

Wir haben bereits in Kapitel 3 und 4 mögliche Ansätze erwähnt, die in zukünftigen Analysen untersucht werden könnten. Unter denkbaren Fragestellungen erachten wir die im folgenden beschriebenen als die interessantesten.

Wir haben gezeigt, dass die Idee der unintuitiven Züge am Beispiel des *Rush-Hour*-Spiels einen vielversprechenden Ansatz darstellt, wenn man sie im Kontext des Problemraums betrachtet. Jedoch muss bei ihrer Betrachtung bedacht werden, dass Menschen bei der Lösung eines Problems vermutlich einige Schritte vorausplanen und unintuitive Züge nur in einigen Fällen sinnvoll sind. Eine systematische Analyse der Problemräume und der Navigation der Versuchspersonen in ihnen mit Augenmerk auf unintuitive Züge ist unumgänglich, um die Hypothese zu überprüfen, dass Versuchspersonen unintuitive Züge meiden und dafür sogar Umwege in Kauf nehmen, zu überprüfen.

Während der Bewertung unserer Komplexitätsmaße und unserer Versuchsergebnisse haben wir einige Schwächen unserer Maße feststellen können. Beispielsweise sind einige Maße nur für den optimalen Graphen definiert, der jedoch von einem Großteil der Versuchspersonen bald verlassen wird. Eine erste Verbesserung könnte hier sein, einen erweiterten optimalen Graphen zuzulassen, der Lösungswege, die 1, 2 oder mehr Schritte von Knoten des optimalen Graphen enthalten. Damit erreichen wir eine höhere Toleranz gegenüber den Fehlern von Versuchspersonen. In diesem Zusammenhang könnte in weiteren Arbeiten erforscht werden, in welchen Zuständen die Versuchspersonen den optimalen Weg verlassen oder Fehler machen. Gibt es Eigenschaften, die diese Spielzustände auszeichnen und von den übrigen unterscheiden?

Auch die Definition der in Kapitel 4 vorgestellten *branching complexity* könnte erweitert werden, sodass sie für den gesamten Zustandsraum definiert ist. Ein weiteres Maß, das möglicherweise die Komplexität des Problemraums beeinflusst, könnte auf der Beobachtung basieren, dass es Zustandsräume mit sogenannten *bridges* gibt, also Kanten, durch deren Entfernen die Anzahl der Zusammenhangskomponenten des Graphs erhöht würde. Besonders, wenn diese Kanten in den Lösungswegen enthalten sind, lässt dies vermuten, dass sie für die Komplexität eines Spiels eine Rolle spielen.

Neben der Analyse der Lösungen der Spieler haben wir in dieser Arbeit Erkenntnisse über das Spielverhalten der einzelnen Spieler zu gewonnen. Bisher haben wir zum Beispiel die Überlegungszeiten aggregiert nach Level, Spielzustand oder Schwierigkeitseinschätzung betrachtet. Eine interessante Fragestellung wäre hier, ob man, wenn die Überlegungszeiten nach Spielern aggregiert sind, eine Differenzierung in mehrere Spielertypen vorgenommen werden kann. Es wäre zum Beispiel denkbar, dass es Spieler gibt, die zunächst die Startkonfiguration betrachten, sich mögliche Lösungswege überlegen und diese dann „nur“ noch ausführen, während andere sofort anfangen, loszuklicken und einen *trial-and-error*-Ansatz verfolgen. Spannend wäre herauszufinden, wieviele Züge die Spieler des ersten

Typs fähig sind voranzuplanen, bevor sie wieder eine Überlegpause einlegen. Eine andere Frage ist, ob die Spieler tatsächlich das Problem in Teilprobleme aufteilen und konsequent zunächst an der Lösung dieses Teilproblems arbeiten, bevor sie sich dem nächsten Teilproblem zuwenden. Fraglich ist auch, ob dies überhaupt in dieser strikten Einteilung möglich ist.

Wir haben zudem gesehen, dass die betrachteten Problemräume überraschend groß sein können. Doch ob wirklich alle enthaltenen Zustände relevant sind, ist nicht direkt ersichtlich. In vielen Fällen macht es keinen Unterschied, ob ein Auto um zwei oder drei Felder bewegt wird, wenn in beiden Fällen die selben Autos blockiert werden und die folgenden möglichen Züge identisch sind. Idee ist hier, äquivalente Zustände zusammenzufassen und den reduzierten Zustandsraum zu betrachten, wie bereits von Wiesberg, Zweig und Reinelt vorgeschlagen.

### 5.3 RELEVANZ DER RESULTATE

Durch die in dieser Arbeit vorgestellte Analyse von Komplexitätsmaßen für das Spiel *Rush Hour* ist die Grundlage gelegt, Maße auszuwählen, denen vertraut werden kann. Weiterhin können die vorliegenden Maße ausgefeilt werden, um die Komplexität des Spiels noch besser wiederzugeben. Diese Analyse kann sowohl theoretischen als auch praktischen Nutzen finden: zum einen kann sie dem Hersteller Vorschläge liefern, um die Startkonfigurationen besser zu kategorisieren. Zum anderen kann sie zum allgemeinen Verständnis beitragen, was Menschen in diesem Zusammenhang als komplex empfinden.





---

## LITERATURVERZEICHNIS

---

- [1] Psychospielchen: Rush Hour.  
[https://psychospielchen.cs.uni-kl.de/index.php/Rush\\_Hour](https://psychospielchen.cs.uni-kl.de/index.php/Rush_Hour).
- [2] ANDERSON, J. R. Problem solving and learning. *American Psychologist* 48 (1993).
- [3] ANDERSON, J. R. *Kognitive Psychologie*, 6. Auflage. Spektrum Akademischer Verlag, 2007.
- [4] CORMEN, T. H., LEISERSON, C. E., RIVEST, R. L. UND STEIN, C. *Introduction to Algorithms*, 3. Auflage. The MIT Press, 2009.
- [5] DÖRNER, D. *Die Logik des Mislingens - Strategisches Denken in komplexen Situationen*, 10. Auflage. rororo, 2011.
- [6] DALGAARD, P. *Introductory Statistics with R*, 2. Auflage. Springer, 2008
- [7] FUNKE, J. Dynamic systems as tools for analysing human judgement. *Thinking and Reasoning* 7 (2001), 69–89.
- [8] FUNKE, J. *Problemlösendes Denken*. Kohlhammer, 2003.
- [9] HALFORD, G. S., WILSON, W. H. UND PHILLIPS, S. Processing capacity defined by relational complexity: implications for comparative, developmental, and cognitive psychology. *Behavioral & Brain Sciences* 21 (1998), 803–865.
- [10] HARTIGAN, J. A. UND WONG, M. A. A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)* 28, 1 (1979), 100–108.
- [11] MILLER, G. The magical number seven plus or minus two: Some limits on our capacity for processing information. *Psychological Review* 63 (1956), 81–97.
- [12] NEWELL, A. Reasoning, problem solving and decision processes: the problem space as a fundamental category. *Computer Science Department Paper 2482* (1979).
- [13] NEWELL, A. UND SIMON, H. A. *Human Problem Solving*. Prentice Hall, Englewood Cliffs, 1972.
- [14] QUESADA, J., KINTSCH, W. UND GOMEZ, E. Complex problem-solving: a field in search of a definition? *Theoretical Issues in Ergonomics Science* 6, 1 (2005), 5–33.
- [15] RAGNI, M., STEFFENHAGEN, F. UND FANGMEIER, T. A structural complexity measure for predicting human planning performance. *Proceedings of the 33rd Annual Conference of the Cognitive Science Society* (2012).

- [16] SUDARSHAN IYENGAR, S. R., VENI MADHAVAN, C. E., ZWEIG, K. A. UND NATARAJAN, A. Understanding human navigation using network analysis. *Topics in Cognitive Science* 4, 1 (2012), 121–134.
- [17] TURING, A. M. On computable numbers, with an application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society* 42 (1936).
- [18] WIRTH, J. UND KLIEME, E. Computer-based assessment of problem solving competence. *Assessment in Education* 10, 3 (2003).
- [19] KOTOVSKY, K., HAYES, J. R. UND SIMON, H. A. Why Are Some Problems Hard? Evidence from Tower of Hanoi. *Cognitive Psychology* 17 (1985).

---

## ABBILDUNGSVERZEICHNIS

---

Abbildung 1	Beispielkonfiguration . . . . .	9
Abbildung 2	Startkonfiguration eines Levels mit Problemraum und optimalem Graphen . . . . .	21
Abbildung 3	Boxplots der Startkonfigurationen nach Schwierigkeitskategorie . . . . .	24
Abbildung 4	Boxplots der Startkonfigurationen nach Editionen . . . . .	26
Abbildung 5	Heatmaps der Korrelationen der Komplexitätsmaße . . . . .	28
Abbildung 6	<i>k-means-clustering</i> der Startkonfigurationen auf Basis ihrer Komplexitätsmaße . . . . .	29
Abbildung 7	Design des Applets . . . . .	31
Abbildung 8	Informationen über die Spieler . . . . .	36
Abbildung 9	Anteil der gelösten, optimal gelösten und abgebrochenen Level . . . . .	37
Abbildung 10	Lösungen der Spieler: benötigte Züge und benötigte Zeit	38
Abbildung 11	Zusammenhang zwischen unseren Maßen und der benötigten Spielzeit sowie den benötigten Zügen . . . . .	40
Abbildung 12	Darstellung des Zustandsraumes für das Level 1 19I . . . . .	41
Abbildung 13	Zusammenhang zwischen Pfadlänge und Level bzw. Schwierigkeitseinschätzung der Spieler. . . . .	42
Abbildung 14	<i>Node visitations</i> . . . . .	44
Abbildung 15	Zustandsraum von Level 1 19I mit zusätzlichen Informationen . . . . .	46
Abbildung 16	Überlegungszeiten der Spieler pro Knoten . . . . .	47
Abbildung 17	Verteilung der Überlegungszeiten der Spieler vor intuitiven und unintuitiven Zügen . . . . .	49
Abbildung 18	Zusammenhang zwischen empfundener Schwierigkeit und unintuitiven Zügen. . . . .	50
Abbildung 19	Ausschnitt aus dem Zustandsraum zu Level 1 10B. . . . .	52
Abbildung 20	Ausgewählte Zustände aus dem Zustandsraum von Level 1 10B. . . . .	53

---

## TABELLENVERZEICHNIS

---

Tabelle 1	Anzahl der verwendeten Startkonfigurationen . . . . .	13
Tabelle 2	Zusammenfassung der Komplexitätsmaße . . . . .	18
Tabelle 3	Mittelwert un Standardabweichung der Komplexitätsmaße	23
Tabelle 4	Maße der für das Experiment ausgewählten Spiele . . .	34
Tabelle 5	Absolute Zahlen der Versuchsergebnisse . . . . .	35
Tabelle 6	Mittelwert, Median, Standardabweichung, Maximalwert und Minimalwert der Überlegungszeiten der Spieler vor intuitiven und unintuitiven Zügen . . . . .	48
Tabelle 7	Nähere Betrachtung der Spielerentscheidungen zu Abbil- dung 19 . . . . .	54

---

ERKLÄRUNG

---

Ich versichere, dass ich diese Bachelorarbeit selbstständig verfasst und nur die angegebenen Quellen und Hilfsmittel verwendet habe.

*Heidelberg, 04. März 2013*

---

Mareike Bockholt