

Albert-Ludwigs-Universität Freiburg
Fakultät für Angewandte Wissenschaften
Institut für Informatik

Studienarbeit

Räumliches Schließen mit Negation

STEFAN SCHLEIPEN

13. November 2007

Betreut von

PROF. DR. BERNHARD NEBEL
NEBEL@INFORMATIK.UNI-FREIBURG.DE

MARCO RAGNI
RAGNI@INFORMATIK.UNI-FREIBURG.DE

Inhaltsverzeichnis

1	Einleitung	6
2	Theoretischer und Mathematischer Ansatz	9
2.1	Theoretischer Ansatz	9
2.2	Mathematischer Ansatz	10
2.2.1	Erstes System	10
2.2.2	Zweites System	12
2.2.3	Drittes System	13
3	Empirische Daten	14
3.1	Erstes Experiment - Akzeptanz	14
3.1.1	Teilnehmer	14
3.1.2	Material	14
3.1.3	Durchführung	15
3.1.4	Design	15
3.1.5	Ergebnisse	15
3.2	Zweites Experiment - Einfaches Generierungsexperiment	16
3.2.1	Teilnehmer	16
3.2.2	Material	19
3.2.3	Durchführung	19
3.2.4	Design	19
3.2.5	Ergebnisse	19
3.3	Drittes Experiment - Komplexität	21
3.3.1	Teilnehmer	21
3.3.2	Material	22
3.3.3	Durchführung	22
3.3.4	Design	22
3.3.5	Ergebnisse	22
3.4	Diskussion	24
4	Implementierung	26
4.1	generelle Konzeptionen	26
4.1.1	Negation Allgemein	26
4.1.2	Inspektion und Variation	26
4.2	Konkrete Implementierung	30
4.2.1	Grundlegende Änderungen	30
4.2.2	Konstruktion	30
4.2.3	Inspektion	33
4.2.4	Variation	35
4.3	Screenshots	39

5	Ergebnisse	43
5.1	Experimente	43
5.2	SRM	43
5.3	Offene Fragen	44

Zusammenfassung

Räumliches Schließen gehört sicherlich zu einem der wichtigen Prozesse des alltäglichen Lebens, und wird beispielsweise bei der Orientierung in bekannten oder unbekanntem Umgebungen, oder bei der zwischenmenschlichen Kommunikation über räumliche Konfigurationen benötigt.

Es wurde bereits eine Menge an beeindruckenden Erkenntnissen in diesem Bereich gewonnen. Als 'Vater' der mentalen Modelle sei hier nur Philip Johnson-Laird erwähnt, der unter anderem die *mental model theory* entwickelt hat. Auch die Arbeiten von Markus Knauff, Reinhold Rauh und Marco Ragni, die die Existenz eines Präferierten Mentalen Modells, kurz PMM, erforscht haben, sollen hier erwähnt werden.

Mit diesen Arbeiten geht einher, dass sie sich nur mit Prämissen beschäftigt haben, die keinerlei Negation enthalten. Zwar gibt es einige Arbeiten über den Begriff der Negation von Barbara Kaup oder auch Uri Hasson und Sam Glucksberg, diese haben sich aber ausschließlich mit Negation im sprachlichen Verständnis beschäftigt.

Ziel dieser Arbeit ist es, den Einfluss von Negation auf den räumlichen Deduktionsprozess zu untersuchen. Die Fragen, die im folgenden geklärt werden sollen, sind: Gibt es PMMs beim räumlichen Schließen mit Negation? Wie sehen diese PMMs aus? Sind sie stabil bei mehrdimensionalen Modellen? Und ist das räumliche Schließen mit Negation komplexer bzw. schwerer als das räumliche Schließen ohne Negation? Der Einfluss der Negation wird sowohl anhand deterministischer als auch indeterministischer Aufgabenstellungen untersucht.

Diese Arbeit unterteilt sich in einen theoretischen, einen experimentellen und einen implementellen Teil. Im theoretischen Teil wird das Problem zunächst auf einer formalen Ebene beschrieben und unterschiedliche Interpretationsmodelle verglichen. Daraus resultierten Fragestellungen die mit Hilfe formaler Methoden nicht zu beantworten sind. Daher wurden drei empirische Untersuchungen entwickelt und durchgeführt. Im experimentellen Teil werden diese Untersuchungen und die daraus gewonnenen Ergebnisse vorgestellt. Damit war es möglich die Theorie Präferierter Mentaler Modelle um das Schließen mit negierten Ausdrücken zu erweitern. Im letzten Teil werden diese empirisch gestützten Erkenntnisse dazu benutzt eine Erweiterung der SRM [RS06] zu implementieren, die es erlaubt, Modelle mit Negation von der SRM verarbeiten zu lassen.

Teile dieser Arbeit wurden bereits auf den Konferenzen KI 2007 und CogSci 2007 vorgestellt und sind unter [SRF07] und [RFS07] veröffentlicht.

1 Einleitung

Die *mental model theory*, die als Grundlage dieser Arbeit dient, wird durch eine große Anzahl von Arbeiten gestützt. Die zu Grunde liegende Idee hierbei ist, dass Menschen beim räumlich Schließen, die räumlichen Relationen der realen oder einer fiktiven Welt in ein mentales Modell übersetzen. Diese Relationen wiederum werden dann benutzt, um räumliche Inferenz zu lösen. Im folgenden ein Beispiel, um das Problem leichter zu verdeutlichen [MJL82]:

Der Löffel ist links von dem Messer.
Der Teller ist rechts von dem Löffel.
Die Gabel ist unter dem Löffel.
Die Tasse ist unter dem Messer.

Anhand der oben genannten Prämissen lassen sich zwei mögliche Modelle erzeugen. Ein solches Modell wird als indeterminiert bezeichnet, da die Prämissen kein eindeutiges, sondern mehrere mögliche Modelle beschreiben.

Löffel	Teller	Messer	Löffel	Messer	Teller
Gabel		Tasse	Gabel	Tasse	

Angenommen ein Kind hilft seiner Mutter beim Decken des Tisches. Das Kind nimmt ein Messer und will es links neben den Teller legen. Als die Mutter dies bemerkt und zu ihm sagt, "Das Messer gehört nicht links neben den Teller". Wo wird das Kind nun das Messer platzieren? Wenn man von einer logischen Interpretation ausgeht, kommen drei mögliche Positionen in Frage, rechts, oberhalb und unterhalb, geht man davon aus, dass die Positionierung auf oder unter dem Teller nicht in Betracht gezogen wird.

Nun stellt sich direkt die Frage: Wie werden solche Probleme verarbeitet? Gibt es eine präferierte Interpretation? Die *mental model theory* (MMT) von Johnson-Laird [JLB91] besagt, dass Menschen in solchen Fällen eine Konklusion nach der anderen erzeugen und diese dann in einem räumlichen Array inspizieren, das den Zustand, der aus den Prämissen erzeugt wurde, repräsentiert. Dieser Prozess besteht aus drei Phasen: dem Verstehen, dem Beschreiben und der Validation. Während der Verstehensphase wird das mentale Modell aufgebaut, welches die Informationen aus den gegebenen Prämissen widerspiegelt. Neue Informationen, die während dem Lesen der Prämissen extrahiert wurden, werden direkt in der Verstehensphase in die Konstruktion des mentalen Modells aufgenommen. In der Beschreibungsphase wird das mentale Modell, das in der Verstehensphase aufgebaut wurde, inspiziert. Die Inspektion dient dazu, neue Informationen aus dem Modell zu extrahieren, die nicht direkt aus den Prämissen zu lesen sind. In der Validationsphase, werden alternative Modelle gesucht, welche die Prämissen erfüllen. Durch den kompletten Prozess bleiben dennoch ein paar Fragen offen. Welches Modell wird beispielsweise als erstes erzeugt? Und, wenn es ein solches initiales Modell gibt, nach welchen Kriterien entsteht dieses Modell, welche

Informationen werden mit dem Modell mitgeführt? Warum werden manche Modelle von Menschen nicht erzeugt?

Diese Fragen können mit der klassischen *mental model theory* nicht ausreichend beantwortet werden. Ein Versuch, sich diesem Problem zu nähern und Antworten auf die oben gestellten Fragen zu finden, stellt die Präferierte Mentale Modell Theorie oder kurz PMMT dar. Die PMMT unterscheidet sich von der MMT in einem wesentlichen Punkt. Um ein Modell auf Konsistenz der Prämissen zu testen, wird in der Variationsphase nicht einfach ein alternatives, mögliches Modell gesucht, sondern initial ein präferiertes Modell erzeugt. Dieses präferierte mentale Modell dient als Ausgangsmodell in der Validation, um durch kontinuierliche Variation des Modells ein Gegenbeispiel zu erzeugen [RHK⁺05].

Bei der Konstruktion des PMM kommen zunächst jedoch verschiedene mögliche Prinzipien der Erzeugung in Frage. Bei der Verarbeitung der Prämissen “B ist links von A” und “C ist links von A” kommen zwei mögliche Prinzipien in Frage: Zum einen das *first fit*-Prinzip, kurz *ff*, und das *first free fit*-Prinzip, kurz *fff*. Im Fall der beiden Prämissen würde zunächst B links neben A platziert. Nach dem Lesen der zweiten Prämisse würde im Fall des *ff*-Prinzips nun das C direkt links neben A gesetzt und das B um eine Position nach links verdrängt. Das resultierende Modell sähe dann so aus: **B C A**. Wird jedoch das *fff*-Prinzip verwendet, so wird wie oben erst das Modell **B A** erzeugt. Beim Lesen der zweiten Prämisse wird jedoch das C nicht zwischen A und B gesetzt, sondern auf die erste freie Position die links von A ist. Das nach dem *fff*-Prinzip erzeugte Modell sieht dann wie folgt aus: **C B A**. Es gibt noch ein weiteres Prinzip das so genannte *mixed fit*-Prinzip. Dabei wird ein Objekt irgendwo zwischen die Position direkt neben dem anderen Objekt der Prämisse und der ersten freien Position platziert. Dieses Prinzip kann natürlich nur zum Tragen kommen, wenn es noch weitere Möglichkeiten der Platzierung eines Objektes gibt.

Die Arbeiten von Ragni, et al. [RFWKed] konnten zeigen, dass beim Konstruieren des PMM das *fff*-Prinzip verwendet wird. Es scheint, dass dieses Prinzip das ökonomischste der oben genannten Prinzipien darstellt. Dies ist wichtig, da das Arbeitsgedächtnis nur über begrenzte Kapazitäten verfügt und dadurch mehr Information behalten werden kann.

Genauere Erkenntnisse darüber wie die Verarbeitung von Negation bei Menschen repräsentiert wird, gibt es bisher keine. Es gibt jedoch einige Arbeiten, die sich mit dem Thema der Negation beschäftigen haben. Barbara Kaup et al. [KLZ06] haben sich mit Negation in Sprache beschäftigt. Dabei wurde ein Verifikationsexperiment durchgeführt, bei dem die Versuchspersonen Situationen mit Hilfe von Sätzen beschrieben bekamen (z.B. “Die Tür ist nicht geöffnet”). Zusätzlich wurden ihnen Bilder präsentiert, die ebenfalls eine Situation zeigten, die entweder identisch mit der des Satzes oder kontrovers war (z.B. eine geöffnete Tür). Die Ergebnisse zeigten, dass die Reaktionszeiten kürzer waren, wenn die Szene auf dem Bild mit dem gesprochenen Satz übereinstimmten. Die Untersuchungen bezogen sich dabei aber immer auf Situationen, die nur zwei mögliche Zustände hatten (Tür: offen, geschlossen). Fälle mit mehreren Möglichkeiten wurden bisher noch nicht untersucht.

1 Einleitung

Hasson und Glucksberg [HG06] haben ebenfalls Untersuchungen im Bereich der Negation gemacht. Sie untersuchten den Unterschied zwischen dem Verstehen von positiven und negativen Behauptungen in der natürlichen Sprache. Die Versuchspersonen mussten dabei lexikalische Entscheidungen in Bezug auf Begriffe, entweder aus positiven oder negativen Bedeutungen, treffen. Die Ergebnisse deuten darauf, dass die negativen Aussagen mental durch äquivalente, positive Aussagen, ersetzt wurden.

Auch in der klassischen Logik bzw. der künstlichen Intelligenz ist die Negation ein wichtiger Bestandteil. So ist in der Logik die Negation einer Relation durch die Konjunktion der verbleibenden Relationen definiert. Die Frage ist allerdings, ob diese Interpretation auch im Falle des menschlichen Schließens zum tragen kommt. Diese Frage kann nur durch Experimente mit menschlichen Versuchspersonen geklärt werden.

Das Ziel dieser Arbeit ist, Probleme mit negierten Relationen zu untersuchen, die Erkenntnisse empirisch zu bestätigen und diese dann formal in die bereits existierende SRM zu implementieren. Für die Implementation wird ein Algorithmus entwickelt der dann in den Konstruktions-, Validations- und Variationsprozess integriert wird.

2 Theoretischer und Mathematischer Ansatz

2.1 Theoretischer Ansatz

Vor den Experimenten mußten noch einige Fragen geklärt werden: Wie würde ein Mensch die Negation einer Prämisse interpretieren? Wie würde er sich Negation merken? Im Folgenden werden deshalb drei Prinzipien vorgestellt, wie eine mögliche Interpretation aussehen könnte. Danach, im mathematischen Teil, werden formale Beschreibungen für die Interpretation von Relationen und der Negation dieser Relationen gegeben.

Die erste hier vorgestellte Interpretationsmöglichkeit ist die konverse Relation anstatt der ursprünglich gegebenen zu verwenden. Dazu wird die Prämisse, die eine negierte Relation enthält, ersetzt durch eine neue Prämisse welche die selben Objekte enthält, aber die Relation ersetzt wurde. Konverse Relation bedeutet hierbei, dass die innerhalb der Dimension gegenüberliegende Relation verwendet wird. So würde "A ist nicht links von B" durch "A ist rechts von B" ersetzt. Wäre diese Interpretation zutreffend, dürfte ein Mensch nicht in der Lage sein, alle konsistenten Modelle zu erzeugen, die durch die Prämissen beschrieben werden. Man würde zum Beispiel erwarten, dass Variation in eine andere Dimension nicht vorkommen wird.

Die zweite mögliche Interpretation wäre, dass die Prämisse unverändert übernommen wird und das zu platzierende Objekt an genau der Stelle platziert wird, die durch die Relation verboten wird. Das Objekt wird dann zusätzlich mit einer Annotation versehen, die die aktuelle Position als nicht zulässig markiert. Diese Methode würde es ermöglichen alle korrekten möglichen Modelle zu erzeugen. Das Erzeugen eines korrekten Modells würde mit dieser Methode aber länger dauern, da jedes Mal, ausgehend von dem falschen Modell, eine Variation durchgeführt werden müßte, um ein korrektes Modell zu erhalten. Diese Tatsache würde die Komplexität des Modells jedoch stark erhöhen. Ob diese Interpretation die menschliche widerspiegelt, könnte demnach leicht in einem Experiment bestimmt werden, indem man negierte Modelle mit nicht negierten vergleicht. Das so erzeugte initiale Modell wäre auch ein Hinweis auf die Existenz von PMMs, wenngleich das Modell inkorrekt wäre.

Die dritte Möglichkeit ist ein Konsens zwischen den bereits vorgestellten Interpretationen. Die Prämisse wird dabei, wie auch im ersten Fall, durch eine neue Prämisse ersetzt, die die konverse Relation enthält. Zusätzlich wird das platzierte Objekt mit einer Annotation wie im zweiten Fall versehen. Diese Interpretation verbindet die Vorteile beider Möglichkeiten ohne jedoch die Nachteile zu übernehmen. Das initiale Modell ist ein korrektes. Durch die Variation mit Hilfe der Annotation wäre ein Mensch in der Lage alle alternativen, korrekten Modelle zu finden. Durch dieses Prinzip sollte auch sich die Komplexität verringern, was für ein ökonomisches Prinzip spricht.

In den vorgestellten Interpretationen wird davon ausgegangen, dass initial ein PMM generiert wird: im zweiten Fall ein inkorrektes, in den anderen beiden ein korrektes. Eine Interpretation ohne ein solches initiales Modell gilt als unwahrscheinlich, da solche Konzepte zu komplex und damit Negation vermutlich nicht lösbar wäre. Der

Nachweis eines PMM aber würde die Vermutung nahe legen, dass das Schließen mit negierten Prämissen nicht komplexer als das Schließen mit indeterminierten Modellen ist. Auch würden dann vermutlich die selben Erkenntnisse, die für das Schließen mit indeterminierten Modellen gefunden wurden, für die Negation gelten.

2.2 Mathematischer Ansatz

Der mathematische oder auch formale Ansatz bezieht sich auf die logische Interpretation der Relationen bei mentalen Modellen. Dazu müssen aber erst einmal die Basisrelationen *links*, *rechts*, *oben*, *unten* definiert werden. Was zum Beispiel bedeutet die Prämisse "A ist über B"? Im Folgenden werden vier Systeme vorgestellt, die eine jeweils unterschiedliche Interpretation beinhalten. In Abbildung 1 sind die vier Systeme dargestellt, die näher behandelt werden. Die ersten beiden Systeme sind Koordinaten basiert, wobei das erste einen strikten Ansatz und das zweite einen kompletten Ansatz darstellt. Die beiden anderen Systeme sind Winkel basiert und unterscheiden sich lediglich in der Aufteilung des Raums. Allen Systemen ist gemein, dass sie relativ für ein Objekt gelten und nicht global. Das Zentrum stellt dabei die Position des relativen Objektes dar und die umliegenden Bereiche sind Positionen für ein lokalisiertes Objekt.

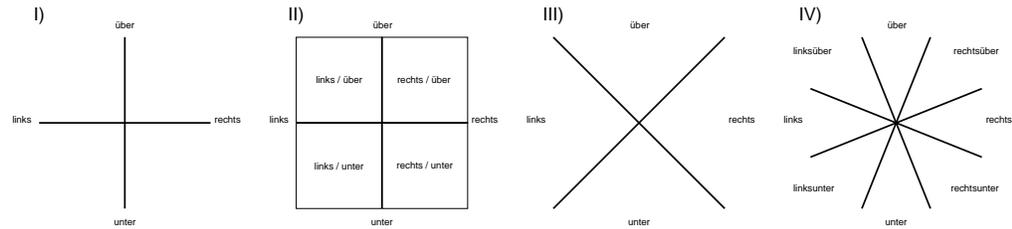


Abbildung 1: Vier mögliche Systeme I) striktes System, II) komplettes System, III) Winkelbasiertes System, IV) System von Gapp [Gap95].

2.2.1 Erstes System

Das erste System basiert auf einem strikten Ansatz (Abb. 1). Die Regeln für die Platzierung eines Objektes gemäss einer Relation sind wie folgt definiert:

$$\begin{aligned}
 \beta \models \text{über}(X,Y) &\Leftrightarrow \beta(x') = \beta(y') \wedge \beta(x'') > \beta(y'') \\
 \beta \models \text{unter}(X,Y) &\Leftrightarrow \beta(x') = \beta(y') \wedge \beta(x'') < \beta(y'') \\
 \beta \models \text{links}(X,Y) &\Leftrightarrow \beta(x') < \beta(y') \wedge \beta(x'') = \beta(y'') \\
 \beta \models \text{rechts}(X,Y) &\Leftrightarrow \beta(x') > \beta(y') \wedge \beta(x'') = \beta(y'')
 \end{aligned}$$

Wobei X ein Tupel (x', x'') ist, mit x' als Position auf der X-Achse und x'' als Position auf der Y-Achse. β ist die Wahrheitsbelegung. Für die Definitionen der negierten

Prämissen wird zunächst die Wahrheitsbelegung benötigt. Diese ist im Falle der vier gegebenen Relationen wie folgt definiert:

$$\beta \models \top \Leftrightarrow (\text{über}(X,Y) \models \top) \oplus (\text{unter}(X,Y) \models \top) \oplus (\text{links}(X,Y) \models \top) \oplus (\text{rechts}(X,Y) \models \top)$$

Das exklusiv-ODER (symbolisch \oplus) bedeutet, dass nur jeweils eine der Relationen gleichzeitig wahr sein darf. Fälle bei denen keine der Relationen oder mehr als eine Relation wahr sind, bedeuten das der gesamte Ausdruck Falsch ist. Die Verwendung des exklusiv-ODER ist deshalb bei der Negation so wichtig, da bei den Modellen immer nur eine Position für das Objekt benutzt werden darf. Eine Platzierung an zwei Positionen zugleich ist nicht zulässig. Das exklusiv-ODER ist definiert als:

$$x \oplus y \Leftrightarrow (x \wedge \neg y) \vee (\neg x \wedge y)$$

Das strikte System erlaubt eine Variation nur entlang der Achsen des euklidischen Raums. Die Areale zwischen den Achsen sind hierbei nicht definiert und fallen als mögliche Positionen für Objekte weg. Die Negation einer Relation erlaubt damit auch nur eine Interpretation entlang der Achsen. Da die Negation genau die Relation verbietet, die in der Prämisse vorkommt, bleiben nur die verbleibenden Relationen als mögliche Positionen übrig. Die Negation ist durch die Disjunktion aller Relationen ($r \in R$, mit R als Menge aller Relationen) ausgenommen der negierten Relation definiert. $\Rightarrow \neg r := (x \mid x \in R_{\setminus \{r\}})$.

Die formale Definition über die Wahrheitsbelegung am Beispiel der Relation $\neg\text{über}(x,y)$:

$$\begin{aligned} \beta \models \neg\text{über}(X,Y) &\Leftrightarrow \neg\text{über}(X,Y) \wedge \top \\ &\Leftrightarrow \neg\text{über}(X,Y) \wedge \\ &\quad ((\text{über}(X,Y) \oplus \text{unter}(X,Y) \oplus \text{links}(X,Y) \oplus \text{rechts}(X,Y))) \\ &\Leftrightarrow (\neg\text{über}(X,Y) \wedge \text{über}(X,Y)) \oplus (\neg\text{über}(X,Y) \wedge \text{unter}(X,Y)) \oplus \\ &\quad (\neg\text{über}(X,Y) \wedge \text{rechts}(X,Y)) \oplus (\neg\text{über}(X,Y) \wedge \text{links}(X,Y)) \\ &\Leftrightarrow (\neg\text{über}(X,Y) \wedge \text{unter}(X,Y)) \oplus (\neg\text{über}(X,Y) \wedge \text{rechts}(X,Y)) \oplus \\ &\quad (\neg\text{über}(X,Y) \wedge \text{links}(X,Y)) \\ &\Leftrightarrow \neg\text{über}(X,Y) \wedge (\text{unter}(X,Y) \oplus \text{links}(X,Y) \oplus \text{rechts}(X,Y)) \\ &\Rightarrow (\text{unter}(X,Y) \oplus \text{links}(X,Y) \oplus \text{rechts}(X,Y)) \end{aligned}$$

Da in einem logischen Ausdruck $\neg \text{über}(X,Y)$ und $\text{über}(X,Y)$ nicht beide gleichzeitig wahr sein können, streichen sich die Relationen gegenseitig. Im Folgenden noch einmal alle Definitionen der negierten Relationen.

$$\begin{aligned}
 \beta \models \neg \text{über}(X,Y) &\Leftrightarrow (\beta(x') = \beta(y') \wedge \beta(x'') < \beta(y'')) \\
 &\quad \oplus (\beta(x') < \beta(y') \wedge \beta(x'') = \beta(y'')) \\
 &\quad \oplus (\beta(x') > \beta(y') \wedge \beta(x'') = \beta(y'')) \\
 \beta \models \neg \text{unter}(X,Y) &\Leftrightarrow (\beta(x') = \beta(y') \wedge \beta(x'') > \beta(y'')) \\
 &\quad \oplus (\beta(x') < \beta(y') \wedge \beta(x'') = \beta(y'')) \\
 &\quad \oplus (\beta(x') > \beta(y') \wedge \beta(x'') = \beta(y'')) \\
 \beta \models \neg \text{links}(X,Y) &\Leftrightarrow (\beta(x') = \beta(y') \wedge \beta(x'') > \beta(y'')) \\
 &\quad \oplus (\beta(x') = \beta(y') \wedge \beta(x'') < \beta(y'')) \\
 &\quad \oplus (\beta(x') > \beta(y') \wedge \beta(x'') = \beta(y'')) \\
 \beta \models \neg \text{rechts}(X,Y) &\Leftrightarrow (\beta(x') = \beta(y') \wedge \beta(x'') > \beta(y'')) \\
 &\quad \oplus (\beta(x') = \beta(y') \wedge \beta(x'') < \beta(y'')) \\
 &\quad \oplus (\beta(x') < \beta(y') \wedge \beta(x'') = \beta(y''))
 \end{aligned}$$

2.2.2 Zweites System

Das zweite System ist eine Erweiterung des ersten zu einem kompletten System (Abb. 1, System II). In diesem System sind die Areale zwischen den Achsen nicht länger undefiniert und es können Objekte auch in diesen Bereichen platziert werden. Mit dieser Definition ist der gesamte euklidische Raum definiert.

$$\begin{aligned}
 \beta \models \text{über}(X,Y) &\Leftrightarrow (\beta(x') \leq \beta(y') \vee \beta(x') > \beta(y')) \wedge \beta(x'') > \beta(y'') \\
 \beta \models \text{unter}(X,Y) &\Leftrightarrow (\beta(x') \leq \beta(y') \vee \beta(x') > \beta(y')) \wedge \beta(x'') < \beta(y'') \\
 \beta \models \text{links}(X,Y) &\Leftrightarrow \beta(x') < \beta(y') \wedge (\beta(x'') \leq \beta(y'') \vee \beta(x'') > \beta(y'')) \\
 \beta \models \text{rechts}(X,Y) &\Leftrightarrow \beta(x') > \beta(y') \wedge (\beta(x'') \leq \beta(y'') \vee \beta(x'') > \beta(y''))
 \end{aligned}$$

Die Areale zwischen den Achsen sind bei dieser Definition aber nicht explizit einer bestimmten Relation zugeordnet. Vielmehr ist die Definition direkt von der Relation innerhalb der Prämisse abhängig. Wenn zum Beispiel die Prämisse “B ist links von A” gegeben ist, sind beide Bereiche links des Zentrums in dem sich A befindet als links definiert (siehe Abb. 1, System II). Beim Lesen der Prämissen in der Konstruktionsphase ist die Definition immer abhängig davon, welche Prämisse gerade verarbeitet wird.

Ebenso wie beim vorherigen System wird durch die Negation nur die Position der Relation selber verboten. Alle verbleibenden Relationen sind erlaubt. Diese Definition der Bereiche wiegt die Negation stärker als eine eventuell andere Prämisse. Das bedeutet, die Bereiche, die durch die Relation definiert werden, sind als Positionen nicht zulässig. Die Definition der negierten Relationen ist ähnlich der des ersten Systems. Zusätzlich sind nun die Bereiche zwischen den Achsen zu beachten. Die Definitionen sind wie folgt:

$$\begin{aligned}
 \beta \models \neg \text{über}(X,Y) &\Leftrightarrow (\beta(x') \leq \beta(y') \oplus \beta(x') > \beta(y')) \wedge \beta(x'') < \beta(y'') \\
 \beta \models \neg \text{unter}(X,Y) &\Leftrightarrow (\beta(x') \leq \beta(y') \oplus \beta(x') > \beta(y')) \wedge \beta(x'') > \beta(y'') \\
 \beta \models \neg \text{links}(X,Y) &\Leftrightarrow \beta(x') > \beta(y') \wedge (\beta(x'') \leq \beta(y'') \oplus \beta(x'') > \beta(y'')) \\
 \beta \models \neg \text{rechts}(X,Y) &\Leftrightarrow \beta(x') < \beta(y') \wedge (\beta(x'') \leq \beta(y'') \oplus \beta(x'') > \beta(y''))
 \end{aligned}$$

2.2.3 Drittes System

Das dritte System (Abb. 1, System III) ist nicht koordinatenbasiert wie die ersten beiden Systeme, sondern Winkel-basiert. Hier sind die Relationen durch aufgespannte Bereiche zwischen zwei Geraden definiert, die sich in bestimmten Winkeln zur X-Achse befinden. Das System ist ähnlich dem von Gapp [Gap95] und wurde von diesem inspiriert (Abb. 1, System IV). Bei seinen Forschungen benutzte er ein Gitter um ein Objekt mit vier verschiedenen Winkeln (0° , 22.5° , 45° , 67.5°) und relativen Abständen vom Mittelpunkt (130, 240, 350, 460 Pixel). Mit diesem Gitter wurde dann getestet wie Menschen bestimmte Positionen akzeptieren oder ablehnen, wenn sie eine Relation dazu präsentiert bekommen. Die Ergebnisse zeigen, dass es dabei Metabereiche gibt, die nicht eindeutig einer bestimmten Relation zugeordnet sind. In unserem System wollen wir aber den gesamten Bereich durch die vier Relationen *oben*, *unten*, *links*, *rechts* definieren. Aus diesem Grund wird auf die Metabereiche verzichtet. Eine Unterteilung in vier Bereiche zu jeweils 90° scheint dabei ausreichend zu sein. Die Definition der einzelnen Relationen:

$$\begin{aligned} \text{über}(x,y) \models \top &\Leftrightarrow \varphi \in [45^\circ, 135^\circ] \\ \text{unter}(x,y) \models \top &\Leftrightarrow \varphi \in [225^\circ, 315^\circ] \\ \text{links}(x,y) \models \top &\Leftrightarrow \varphi \in [135^\circ, 225^\circ] \\ \text{rechts}(x,y) \models \top &\Leftrightarrow \varphi \in [315^\circ, 45^\circ] \end{aligned}$$

φ ist der Winkel zwischen der positiven X-Achse und der Geraden die zwischen dem Objekt im Zentrum und dem in Relation stehenden Objekt verläuft. Bei den negierten Relationen wird, wie bei den anderen Systemen bereits auch, eine Positionierung im Bereich der nicht negierten Relation verboten. Das Objekt kann also nur in einem Bereich, außerhalb eines von der Relation vorgegebenen Bereichs, platziert werden. Die resultierenden Definitionen sehen wie folgt aus:

$$\begin{aligned} \neg \text{über}(X,Y) \models \top &\Leftrightarrow \varphi \in [135^\circ, 45^\circ] \\ \neg \text{unter}(X,Y) \models \top &\Leftrightarrow \varphi \in [315^\circ, 225^\circ] \\ \neg \text{links}(X,Y) \models \top &\Leftrightarrow \varphi \in [225^\circ, 135^\circ] \\ \neg \text{rechts}(X,Y) \models \top &\Leftrightarrow \varphi \in [45^\circ, 315^\circ] \end{aligned}$$

Dieses System kann als sehr wahrscheinlich betrachtet werden, da bereits das Experiment von Gapp [Gap95] gezeigt hat, dass diese Einteilung von Menschen angenommen wird. Ob eine Verfeinerung, wie sie bei Gapp vorliegt, sinnvoll ist, wird sich im Verlauf der Experimente zeigen.

Man darf davon ausgehen, dass wenigstens eines der drei Systeme einen relativ guten Ansatz darstellt, wie Relationen und deren Negation von Menschen interpretiert werden. Weiter kann man davon ausgehen, dass auch bei Negation PMMs verwendet werden. Da ein Objekt nur an einer Stelle positioniert werden darf wird erwartet, dass bei PMMs, die konverse der negierten Relation verwendet wird. Wenn sich dies bewahrheiten sollte, wäre das konstruierte PMM genau das Modell, welches die negierte Prämisse durch eine neue Prämisse mit den jeweils konversen, positiven Relationen ersetzt hätte.

3 Empirische Daten

Im Rahmen der Studienarbeit wurden drei Experimente mit Versuchspersonen durchgeführt. Die Experimente waren alle unterschiedlich ausgelegt, um verschiedene Aspekte genauer untersuchen zu können. Die zentralen Fragen der Experimente waren: (i) Wie werden Objekte in Modellen platziert, wenn die Relation negiert wurde? (ii) Werden beim Erzeugen der Modelle bestimmte Relationen benutzt, um Negation der Relation zu interpretieren, und wenn ja welche Relationen führen zu einem präferierten mentalen Modell in der Konstruktionsphase? (iii) Sind präferierte mentale Modelle, die aus Prämissen mit negierten Relationen entstehen anders als präferierte mentale Modelle, die aus indeterminierten Modellen entstehen? (iv) Welchen Einfluss hat es auf das konstruierte Modell, wenn das Modell von links oder rechts aufgebaut werden muss? (v) Gibt es generelle Unterschiede zwischen indeterminierten und negierten Modellen?

Im Vorfeld wurden einige Überlegungen bezüglich der zu erwartenden Ergebnisse angestellt. Es wurde zum Beispiel davon ausgegangen, dass die Versuchspersonen die negierte Prämisse für die Konstruktion des PMM durch eine andere Prämisse ersetzen. Die Vermutung ist, dass die konverse Relation dazu eingesetzt wird. Anstatt der Prämisse "A ist nicht links von B" würde die Prämisse "A ist rechts von B" verwendet werden, um das PMM zu konstruieren. Weitere Vermutungen sind, dass Modelle mit negierten Prämissen schwerer zu konstruieren sind, als Modelle ohne Negation der Prämissen. Ebenso, dass Modelle mit Negation komplexer sind als Modelle die nur indeterminiert sind.

3.1 Erstes Experiment - Akzeptanz

Ziel dieses Experimentes war es die Frage zu klären, welche Interpretation Menschen benutzen, wenn sie Relationen klassifizieren. Die hieraus resultierenden Ergebnisse sollen Aufschluß darüber geben, welches der in Kapitel 3 vorgestellten Systeme dem des Menschen am nächsten kommt. Dazu mussten die Versuchspersonen eine Aussage, die zusammen mit einem Bild präsentiert wurde, bestätigen oder ablehnen.

3.1.1 Teilnehmer

An diesem Experiment nahmen 36 Studenten der Universität Freiburg teil. Das Experiment wurde in zwei Gruppen durchgeführt: 20 Studenten mit hinterlegtem Gitter (Alter: $M = 24.3$, $SD = 2.4$), 16 Studenten ohne hinterlegtem Gitter (Alter: $M = 24$, $SD = 2.8$).

3.1.2 Material

Die Durchführung fand am Computer statt. Das Experiment wurde mit SuperLab 2.0 erstellt und durchgeführt. Für die Eingabe der Antworten wurde eine Standardtas-

tatur verwendet, bei der alle nicht verwendeten Tasten entfernt wurden um falsche Tastendrucke zu vermeiden.

3.1.3 Durchführung

Vor der Durchführung wurde von den Versuchspersonen ein Fragebogen zur statistischen Datenerhebung ausgefüllt. Nach der Durchführung wurde ein weiterer Fragebogen ausgefüllt, der Fragen bezüglich der Schwierigkeit und persönlichen Sichtweise zu diesem Experiment enthielt. Zur Durchführung wurden die Versuchspersonen einzeln an den PC gesetzt und dort durch textuelle Anweisungen instruiert. Die Aufgaben mussten selbständig und ohne Hilfe durchgeführt werden.

3.1.4 Design

Den Versuchsteilnehmern wurde ein Bild zusammen mit einer Aussage präsentiert. Die Versuchspersonen mussten dann entscheiden, ob die Aussage mit der auf dem Bild zu sehenden Situation ihrer Meinung nach übereinstimmte oder nicht. Die Antworten konnten mit "Ja" oder "Nein" gegeben werden. In Abbildung 2 ist eine der Aufgaben abgebildet, links die Aufgabenversion mit hinterlegtem Gitter und rechts die selbe Aufgabensituation ohne hinterlegtem Gitter. Allen Aufgaben ist gleich, dass das A in der Mitte des Bildes fixiert ist und als relatives Objekt dient. Das B wurde im Verlauf des Experimentes durch alle 48 freien Positionen verschoben. Alle daraus entstehenden Konstellationen wurden zusammen mit den Aussagen, "B ist nicht rechts von A" und "B ist nicht über A" gezeigt. Um einen Vergleich zu den nicht negierten Fällen zu geben wurden ebenfalls die Aussagen "B ist rechts von A" und "B ist über A" abgefragt. Die nicht negierten Aussagen wurden in 16 der 48 möglichen Fälle gezeigt. In Abbildung 4 sind Beispielaufgaben aus dem Experiment abgebildet. Die Bilder Zeigen den Versuchsaufbau einmal mit und einmal ohne hinterlegtem Gitter. Bei diesem Experiment wurden die Antwortgenauigkeit und die Reaktionszeit der Versuchspersonen gemessen.

3.1.5 Ergebnisse

Jeder der Teilnehmer des Experimentes vollzog eine klare Trennung des Gitters, sowohl bei den negierten (*nicht rechts/nicht über*), als auch den nicht negierten Fällen (*rechts / über*). In Abbildung 3 kann man gut erkennen, wie stark die Abgrenzung für die Aussage "B ist über/nicht über A" ist. In beiden Fällen, mit und ohne hinterlegtem Gitter, sind die Ergebnisse gleich für alle vier Aussagen (*über/nicht über* bzw *rechts/nicht rechts*).

Die Reaktionszeiten für die negierten Fälle (mit und ohne hinterlegtem Gitter) zeigen einen signifikanten Unterschied zwischen *nicht über* und *nicht rechts* (mit/ohne Gitter: $t = 7.076/5.589, df = 19/15, p \leq 0.01$), ebenso für die nicht negierten Fälle *rechts* und *über* (mit/ohne Gitter: $t = 3.326/4.062, df = 19/15, p \leq 0.01$). Dies läßt die Vermutung

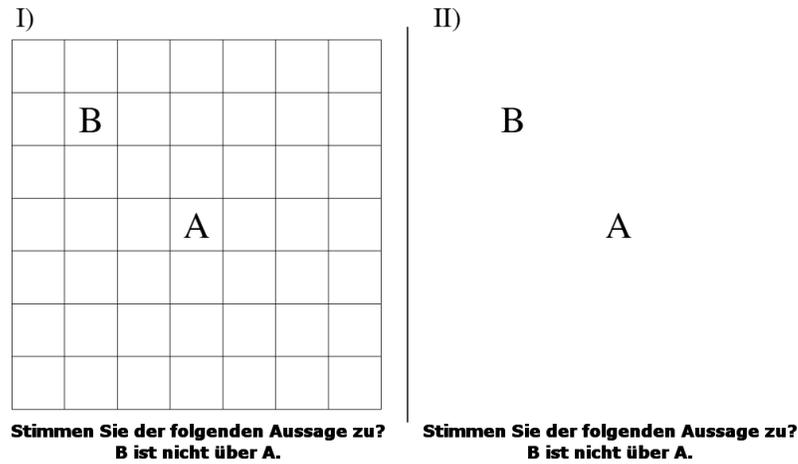


Abbildung 2: Aufgabenstellung aus Experiment 1. Links die Aufgabe mit hinterlegtem Gitter (I), rechts die selbe Aufgabe ohne Gitter (II). Darunter die Aussagen wie sie während des Experimentes zu sehen waren.

zu, dass horizontale Operationen Menschen leichter fallen als vertikale Operationen. Ein weiteres Ergebnis zeigt, dass in den meisten Fällen die Reaktionszeiten signifikant kürzer sind, wenn die Aussage und die Relation von A und B auf dem gezeigten Bild übereinstimmen. Mit hinterlegtem Gitter ergaben sich folgende Werte: *nicht über* $t = 0.288, df = 19, p = n.s./p \leq 0.01$; *nicht rechts* $t = 1.717, df = 19, p = n.s./p \leq 0.05$; *über* $t = 2.810, df = 19, p \leq 0.05/p \leq 0.01$; *rechts* $t = 4.157, df = 19, p \leq 0.001/p \leq 0.05$). Ohne Gitter: *nicht über* $t = 4.124, df = 15, p = n.s./p \leq 0.01$; *nicht rechts* $t = 3.186, df = 15, p = n.s./p \leq 0.05$; *über* $t = 3.550, df = 15, p \leq 0.05/p \leq 0.01$; *rechts* $t = 2.422, df = 15, p \leq 0.001/p \leq 0.05$).

3.2 Zweites Experiment - Einfaches Generierungsexperiment

Bei diesem Experiment wurde untersucht, wie durch Menschen ein Modell, das eine negierte Prämisse enthält, konstruiert wird. Zusätzlich wurde untersucht, ob es präferierte mentale Modelle (PMM) bei der Konstruktion mit negierten Prämissen gibt.

3.2.1 Teilnehmer

An diesem Experiment nahmen 23 Studenten der Universität Freiburg teil. Die Studenten waren zwischen 19 und 36 Jahren alt ($M = 25.8, SD = 4.5$).

3.2 Zweites Experiment - Einfaches Generierungsexperiment

I)

Nein			nicht über			Ja							
100 3256	100 1767	100 2910	93 2482	100 3302	100 2447	100 2312	0	0	0	7	0	0	0
93 2528	100 2616	100 3165	100 1871	100 2578	100 3652	93 2589	7	0	0	0	0	0	7
100 1815	100 2987	100 2628	100 1852	93 2478	100 3412	93 2169	0	0	0	0	7	0	7
7	19	6	A	13	7	19	93 2761	81 3823	94 3936	A	87 2652	93 2173	81 4821
0	7	0	14	0	0	7	100 2997	93 2447	100 2577	86 2260	100 2113	100 2283	93 2462
19	23	6	8	0	7	8	81 2031	77 2106	94 2534	92 1711	100 2086	93 2029	92 2339
8	0	0	0	7	0	6	92 1667	100 1640	100 2550	100 2534	93 2299	100 2375	94 3265

II)

Nein			über			Ja							
	0	7	0	0	7		100 1533	93 2207	100 2053	100 1771	93 2028		
	0				0		100 1855				100 2116		
	93 2265		A		100 2210		7		A		0		
	100 2077				100 1949		0				0		
	100 2558	93 2376	85 1856	94 2331	100 2015		0	7	15	6	0		

Abbildung 3: Die grafische Auswertung des ersten Experimentes. In I) sind die Daten für die Aussage 'nicht über' abgebildet. In den dunkleren Bereichen sind die Prozentangaben für die korrekten Antworten. Die kleineren Zahlen darunter geben die durchschnittliche Reaktionszeit der Teilnehmer für die korrekten Antworten an. In II) sind die Auswertungen für die Aussage 'über' zu sehen. Die Darstellung der Ergebnisse ist identisch mit der für 'nicht über'.

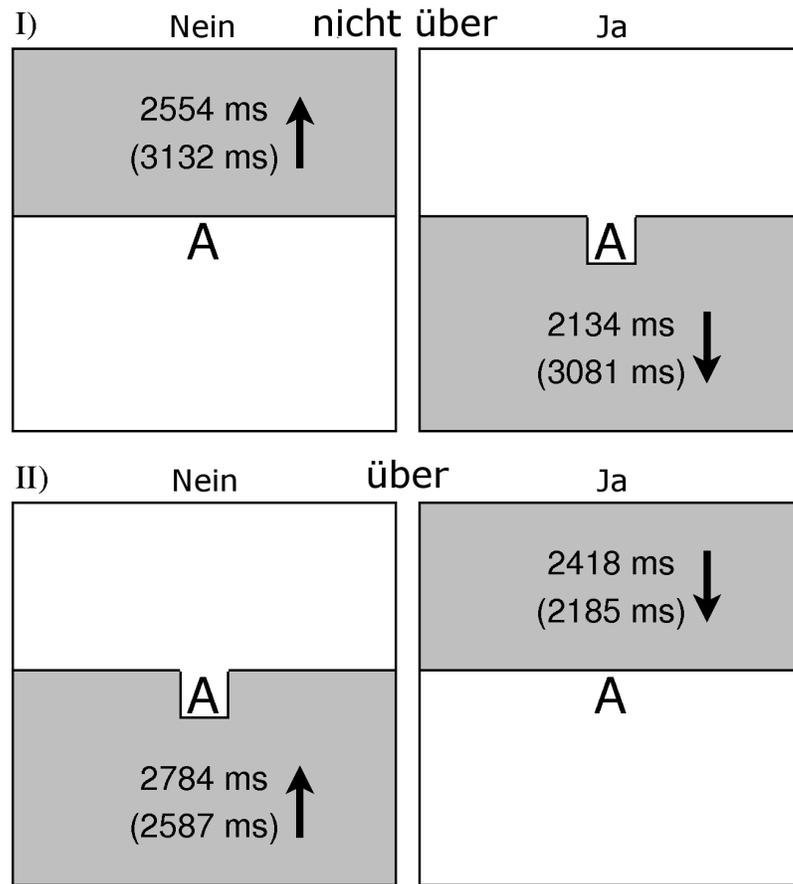


Abbildung 4: Grafische Darstellung der Reaktionszeiten des ersten Experimentes. I) zeigt die durchschnittlichen Reaktionszeiten für eine korrekte Antwort für die Aussage 'nicht über'. Die Zeiten in Klammern stammen aus dem Versuchsaufbau mit hinterlegtem Gitter, die Zahlen darüber ohne ein hinterlegtes Gitter. Die Pfeile zeigen, wie sich die Reaktionszeiten verhalten wenn die Aussage mit der tatsächlich gezeigten Situation übereinstimmen oder nicht. In II) die Daten für die Aussage 'über'. Die Darstellung ist identisch mit der für 'nicht über'.

3.2.2 Material

Das Experiment wurde auf Papier durchgeführt. Die Versuchspersonen bekamen einen gebundenen Stapel Papier, der alle Aufgaben enthielt, und das benötigte Schreibmaterial. Eigene Stifte oder sonstige Hilfsmittel wurden nicht gestattet.

3.2.3 Durchführung

Vor Beginn des Experimentes wurden statistische Daten bezüglich der Versuchsperson mittels eines Fragebogens erhoben. Ebenso wurde nach dem Experiment ein weiterer Fragebogen ausgefüllt, mit dem der Schwierigkeitsgrad des Experimentes beurteilt werden sollte und zusätzlich die persönlichen Erfahrungen durch das Experiment. Im eigentlichen Experiment wurde jeder Versuchsperson ein Exemplar des Versuchs mit dem benötigten Schreibmaterial übergeben. Alle Instruktionen wurden auf Papier gegeben. Darüber hinaus wurde keine Hilfestellung gegeben.

3.2.4 Design

Aufgabe war es, ein Modell aus vier gegebenen Prämissen zu konstruieren und dieses dann auf einem leeren Blatt Papier von Hand zu zeichnen, siehe Tabelle 1. Die Aufgaben wurden in der Dimension (ein- gegen zweidimensional), der Determiniertheit (determiniert gegen indeterminiert) und Negation (negiert gegen nicht negiert) variiert. Jedes Modell wurde zwei mal präsentiert, jedoch mit unterschiedlichen Objekten, um ein Wiedererkennen zu vermeiden. Alle 16 Modelle mussten auf die selbe Weise konstruiert werden. Mittels vier Prämissen wurden fünf Objekte in einem Modell angeordnet. Dabei wurden nur die Relationen *links*, *rechts*, *oben* und *unten* verwendet. Die Negation wurde immer in der dritten Prämisse verwendet, um möglichst störungsfrei die Ergebnisse auswerten zu können. Die Modelle mit Negation waren zwar von der Aufgabenstellung auch in determinierte und indeterminierte Modelle unterteilt. Wegen der Negation sind alle Modelle als indeterminiert zu betrachten. Um zu erreichen, dass die Modelle im Arbeitsgedächtnis konstruiert werden, wurden die Aufgaben auf drei Seiten präsentiert. Die ersten beiden Prämissen auf der ersten Seite die letzten beiden auf der zweiten Seite. Die dritte Seite wurde leer gelassen und diente ausschließlich zum Zeichnen des Modells. Im Falle einer Aufgabe, die mehrere korrekte Modelle zulässt, wurden die Versuchspersonen instruiert nur ein Modell zu zeichnen. Während des Versuchs durften keine Notizen oder Hilfszeichnungen angefertigt werden.

3.2.5 Ergebnisse

Die Menge der korrekten Antworten zeigen, dass eindimensionale Probleme signifikant häufiger korrekt gelöst wurden als zweidimensionale (Wilcoxon-Test: $Z = 3.109$, $p = 0.002$). Weiter gibt es einen signifikanten Unterschied zwischen negierten und nicht negierten Problemen (Wilcoxon-Test: $Z = 2.618$, $p = 0.009$). Keine nachweisbare Signifikanz konnte zwischen determinierten und indeterminierten Problemen gezeigt werden.

Tabelle 1: Die Tabelle zeigt die vier in Experiment 2 verwendeten Aufgaben, jeweils als negierte (I) und nicht negierte (II) Version. Die Modelle sind variiert in der Dimension, ein-dimensional (a, b), zwei-dimensional (c, d) und in der Determiniertheit, determiniert (a, c), indeterminiert (b, d).

Problem	PMM/alternative Modelle
(a) A ist links von B. B ist links von C. <i>I) C ist links von D.</i> <i>II) C ist nicht rechts von D.</i> D ist links von E.	(1) A B C D E
(b) A ist links von B. B ist links von C. <i>I) D ist rechts B.</i> <i>II) D ist nicht links von B.</i> D ist links von E.	(1) A B C D E (2) A B D E C (3) A B D C E
(c) A ist über B. B ist links von C. <i>I) C ist links von D.</i> <i>II) C ist nicht rechts D.</i> D ist unter E.	(1) A E B C D
(d) A ist über B. B ist links von C. <i>I) D ist rechts von C.</i> <i>II) D ist nicht links von C.</i> D ist unter E.	(1) A E B C D (1) A E B D C

Tabelle 2: Die Tabelle zeigt die korrekten Antworten (in Prozent) für ein- und zwei-dimensionale, negierte und nicht negierte, und für determinierte und indeterminierte Probleme.

	1-dim		2-dim	
	Aff.	Neg.	Aff.	Neg.
Det.	87	78	76	52
Indet.	85	78	67	57

Tabelle 3: Die Tabelle zeigt die Präferenz für die konverse Relation in Prozent für ein-dimensionale und zwei-dimensionale, sowie determinierte und indetermierte, negierte Probleme. Die Zahlen die durch einen Doppelpunkt getrennt sind zeigen an erster Stelle die Anzahl der korrekten Antworten in denen die konverse Relation verwendet wurde gegen die Anzahl aller korrekten Antworten. Die unterste Zeile zeigt die Anzahl der jeweils verwendeten Strategien, wenn die konverse Relation verwendet wurde. *fff* = *first free fit* - Strategie, *ff* = *first fit* - Strategie. * $p \leq 0.05$, ** $p \leq 0.01$, *** $p \leq 0.001$

		1-dim.	2-dim.
Det.	opposite	81%***	83%**
	opposite: all	29:36	20:24
Indet.	opposite	75%*	65%
	opposite: all	27:36	17:26
	fff / ff	24*** / 2	14*** / 3

Eine weitere Frage, die mit diesem Experiment geklärt werden sollte, war, wie die Versuchspersonen negierte Probleme verstehen. Genauer, wenn eine Relation durch Negation nicht zulässig ist, sind alle anderen Relationen die verbleiben, korrekte Möglichkeiten um das Objekt zu platzieren. Aus den Ergebnissen geht hervor, dass es eine stabile Präferenz für die konverse Relation bei negierten Problemen gibt. Tabelle 3 zeigt, dass bis auf zweidimensionale, indetermierte Probleme die konverse Relation signifikant häufiger für das konstruierte Modell gewählt wurde. Für ein- und zweidimensionale Probleme ergab sich ein signifikanter Unterschied von Null. Keine signifikanten Unterschiede gab es hingegen bei der Analyse der indetermierten Probleme im nicht negierten Fall gegenüber den entsprechenden Problemen mit Negation.

3.3 Drittes Experiment - Komplexität

Im dritten Experiment wurde die Komplexität über PMMs getestet. Die Versuchspersonen mussten ein Modell konstruieren und dann in einem zweiten Schritt prüfen, ob eine gegebene Aussage wahr ist oder nicht.

3.3.1 Teilnehmer

Sechzehn Studenten der Universität Freiburg nahmen an diesem Experiment teil. (Alter: $M = 24.3$, $SD = 2.4$).

3.3.2 Material

Das Experiment wurde am Computer durchgeführt, um Reaktionszeiten messen zu können. Das Experiment wurde mit SuperLab 2.0 erstellt und durchgeführt. Für die Eingabe der Antworten wurde eine Standardtastatur verwendet, bei der alle nicht verwendeten Tasten entfernt wurden um falsche Tastendrucke zu vermeiden.

3.3.3 Durchführung

Vor der Durchführung wurde von den Versuchspersonen ein Fragebogen zur statistischen Datenerhebung ausgefüllt. Nach der Durchführung wurde ein weiterer Fragebogen ausgefüllt, der Fragen bezüglich der Schwierigkeit und persönlichen Sichtweise zu diesem Experiment enthielt. Zur Durchführung wurden die Versuchspersonen einzeln an den PC gesetzt. Die Instruierung der Versuchspersonen erfolgte am PC durch Eingblendete Anweisungen. Der Versuchsleiter gab keine weiteren Instruktionen. Die Aufgaben mussten selbstständig und ohne Hilfe bearbeitet werden.

3.3.4 Design

Bei den Messungen mit dem Computer wurden Reaktionszeiten, Zeit die von den Versuchspersonen zum Lesen der Prämissen benötigt wurde, und die Genauigkeit der Antworten gemessen. Das Experiment bestand aus 20 Aufgaben, zehn ein-dimensionale und zehn zwei-dimensionale (siehe Abb. 4).

Alle vier Prämissen wurden gleichzeitig präsentiert. Nach einem Tastendruck verschwanden die Prämissen und eine Aussage wurde eingeblendet. Die Aussage bezog sich auf zwei Objekte aus dem Modell, wobei die Relation zwischen den Objekten frei gelassen wurde. Ein Objekt in der Aussage war immer aus der dritten Prämisse, welche die Negation oder Indeterminiertheit enthielt. Das zweite Objekt aus einer der anderen drei Prämissen. Dies garantiert, dass die Versuchspersonen das Modell über die negierte, indetermierte bzw. determinierte Relation prüfen mussten. Die Aussage, die den Versuchspersonen gezeigt wurde, formulierte immer eine Prämisse, die nicht bereits in den gegebenen Prämissen enthalten war. Als Antwortmöglichkeit konnte aus den vier bekannten Relationen gewählt werden: *links*, *rechts*, *über*, *unter*. Die Antworten wurden mit dem Drücken einer Taste am Computer gegeben.

3.3.5 Ergebnisse

Zwei der Versuchspersonen wurden aufgrund einer zu geringen Leistung ($< 50\%$) bei den deterministischen Problemen von der Auswertung Ausgeschlossen. Es konnten keine Unterschiede zwischen der Lesezeit der Prämissen bei negierten, indetermierten und determinierten Modellen festgestellt werden. Ebenso wurden keine Unterschiede bei den Lesezeiten bei Form und Dimension der Modelle gefunden. Es konnte jedoch wieder eine signifikante Präferenz für ein PMM festgestellt werden. In 76% der Fälle

Tabelle 4: Die Tabelle zeigt die im dritten Experiment gestellten Aufgaben. Die kursiven Zeilen repräsentieren die verschiedenen Typen von Modellen die verwendet wurden. (3a) negierte Variante, (3b) indetermierte Variante, (3c) determinierte Variante. Eine Hälfte wurde mit der Relation *über* I) in der vierten Prämisse, die andere mit der Relation *unter* II) gestellt.

Problem	PMM/alternative models
(a) A ist über B.	(I) $A \ E$
B ist links von C.	$B \ C \ D$
<i>3a D ist nicht links von B.</i>	
<i>3b D ist rechts von B.</i>	(II) A
<i>3c C ist links von D.</i>	$B \ C \ D$
D ist unter E. (I) / D ist über E. (II)	E
(b) A ist über B.	(I) $E \ A$
B ist rechts von C.	$D \ C \ B$
<i>3a D ist nicht rechts von B.</i>	
<i>3b D links von B.</i>	(II) E
<i>3c C ist rechts D.</i>	$D \ C \ B$
D ist unter E. (I) / D ist über E. (II)	A

in denen ein korrektes Modell erzeugt wurde, wurde ein PMM erzeugt. (indeterminierte und negierte Probleme: PMM = 76%; alternative Modelle = 24%; Binomial-Test $p \leq 0.001$). Die Genauigkeit der Antworten nahm signifikant von determinierten zu indeterminierten zu negierten Problemen ab, statistisch Ausgewertet durch Page-L Test $N = 14, k = 3, L = 178, p \leq 0.05$.

3.4 Diskussion

Die Ergebnisse aller drei Experimente konnten die eingangs gestellten Fragen beantworten. Es wurden sowohl Hinweise auf präferierte mentale Modelle gefunden, als auch eine signifikante Präferenz für die konverse Relation bei der Konstruktion des initialen Modells (Experiment 2, determiniert: 1-dim. 81%, 2-dim. 83%, indeterminiert: 1-dim. 75%, 2-dim. 65%). Die Frage, ob ein Problem mit Negation schwieriger ist als ein deterministisches oder indeterministisches, kann ebenfalls mit ja beantwortet werden (siehe Tabelle 2). So zeigten sich signifikante Tendenzen, dass negierte Probleme anfälliger für Fehler sind als andere Probleme. Dies zeigt, dass Schließen mit Negation für den Menschen schwieriger ist (korrekte Antworten bei indet., 2-dim. Problemen nur 57%). Ebenso konnten Hinweise darauf gefunden werden, dass eine horizontale Relation schneller verarbeitet wird als eine vertikale (für negierte Relationen: mit/ohne Gitter: $t = 7.076/5.589, df = 19/15, p \leq 0.01$ bzw. für positive Relationen mit/ohne Gitter: $t = 3.326/4.062, df = 19/15, p \leq 0.01$). Die Frage ob präferierte mentale Modelle aus negierten Problemen sich von denen indeterminierter unterscheiden, konnte nicht gezeigt werden. Viel mehr sind durch die Interpretation der negierten Relation durch die konverse Relation, diese Modelle mit der Konstruktion eines Modells aus einem indeterminierten Problem vergleichbar. Ob es Unterschiede gibt, wenn ein Modell von links oder rechts aufgebaut wird, konnte in keinem der Experimente gezeigt werden. Ebenso scheint es keine generellen Unterschiede zwischen indeterminierten und negierten Modellen zu geben.

Die Ergebnisse der Arbeiten von Ragni et al. [RFWK06] bzw. [RFWKed] zum Thema präferierter mentaler Modelle, haben bereits gezeigt, dass ein ökonomisches Prinzip bei der Modellkonstruktion verwendet wird. Dass ein solches Prinzip auch bei Negation verwendet wird, war stark anzunehmen. Dass diese Prinzipien jedoch nahezu identisch mit denen indeterministischer Modelle sind, ist eine interessante Erkenntnis. Diese Erkenntnis kann mit denen von Uri Hasson und Sam Glucksberg [HG06] verglichen werden. In ihrer Arbeit konnten sie zeigen, dass Sätze die Negation enthalten, durch Sätze ersetzt werden, die nicht länger die Negation enthielten. Die Aussagen des Satzes wurden durch eine Neuformulierung mit einem nicht negierten Satz, der die selbe Aussage enthielt, ersetzt. Ein dies bezüglich interessanter Punkt ist eine Erkenntnis, die sich im Verlauf der Befragung der Versuchsteilnehmer herauskristallisierte. Bei der Generierung der Modelle waren sich die Teilnehmer ausnahmslos darüber im klaren, dass das in Frage stehende Objekt auch an einer anderen Position hätte platziert werden können, was die Vermutung einer Annotation des Objektes stark untermauert. Es ist dem Menschen in diesem Moment also eindeutig bewusst, dass es keine eindeutige Position für das Objekt gibt. Da die Versuchsteilnehmer gezwungen waren ein Mo-

dell zu Konstruieren mussten sie sich für eine Position entscheiden. Die Erkenntnisse aus dem ersten Experiment, in welchem eine horizontale Relation schneller verarbeitet wird als eine vertikale, können mit dem Umstand erklärt werden, dass es einem Menschen natürlicher ist sich auf einer horizontalen Achse gedanklich zu bewegen, als auf einer vertikalen. Beispielsweise bei der Orientierung, bei der man seine Umgebung nach *rechts* und *links*, bzw *vor* und *hinter* aufteilt, selten jedoch nach *oben* und *unten*. Auch aus dem ersten Experiment wissen wir, dass wenn sich Aussage und dargestellte Situation gleichen, die Reaktionszeiten kürzer sind. Diese Erkenntnis deckt sich mit denen von Kaup [KLZ06]. Auch in dieser Arbeit waren die Reaktionszeiten kürzer, wenn sich Satz und gezeigtes Bild glichen. Übergreifend auf alle Experimente zeigte sich, dass es keine signifikanten Unterschiede zwischen negierten und in-deterministischen Modellen gibt. Dies lässt darauf schließen, dass die ökonomischen Prinzipien beider Fälle nahezu identisch sind, was wiederum auch für die starke Deckung bei genau diesen beiden Arten von Problemen in den Ergebnissen spricht. Letztlich müssen die Ergebnisse jedoch in dem Kontext verstanden werden, dass es sich bei diesen Untersuchungen um eine eingeschränkte Domäne handelte. Die Untersuchungen waren so ausgelegt, dass die Existenz eines präferierten mentalen Modells im Falle eines negierten Problems bewiesen wird. Deshalb sollten die Versuchspersonen stets genau ein Modell erzeugen. Es wurde auch jeweils nur eine negierte Prämisse verwendet und niemals das Objekt, das durch eine negierte Relation positioniert wurde, durch eine zweite Prämisse in Relation gestellt. Bestrebungen diese Fälle auch abzudecken erfordern weitere Untersuchungen, um eindeutig geklärt werden zu können.

Zusammenfassend lautet die Aussage zu den Ergebnissen deshalb: Beim räumlichen Schließen und der Verwendung von Negation in einer der Prämissen wird eine Neuinterpretation der negierten Prämisse vorgenommen und mit dieser neuen Prämisse das präferierte mentale Modell erzeugt. Durch eine Annotation des Objektes wird die Negation jedoch weiterhin im Gedächtnis behalten.

4 Implementierung

4.1 generelle Konzeptionen

4.1.1 Negation Allgemein

Da die SRM bisher über keine Repräsentation der Negation verfügt, sind im Vorfeld der Implementation einige grundlegende Überlegungen notwendig. Die bisherige Vorgehensweise der SRM ist, die Relationen mittels eines Parser über eine XML Datei in das System zu importieren. Die Relationen können dadurch ohne großen Umstand definiert und bei Gelegenheit verändert werden. Jede Relation wird separat für sich mit einer eigenen Datei definiert. Innerhalb der XML Datei sind Informationen über Name, konverse, etc. (siehe, Abb. 5) enthalten. Die negierten Relationen können also über den XML Parser komplett neu durch eine XML Datei definiert werden. Der Vorteil wäre, dass man so die Möglichkeit hat, der SRM verschiedene Interpretationen der Negation zu übergeben. Ein Problem dabei ist, dass die Negation beim Parsen der Prämissen nicht mehr als Information weiter besteht, da die negierten Relationen in die Basisrelationen übersetzt werden.

Es gäbe allerdings die Möglichkeit, die Negation erst in der Konstruktionsphase zu interpretieren. Die Idee dabei ist, die Negation mit der Prämisse mitzunehmen und erst in der Konstruktionsphase die konverse Relation zu bilden und die Negation als Annotation an das Objekt mit anzuhängen. Die Konverse kann dabei aus den Basisrelationen entnommen werden, da sie als Attribut mitgeführt wird (siehe Abb. 5).

4.1.2 Inspektion und Variation

Ein weiterer wichtiger Punkt ist die Variations- und Inspektionsphase. Da die SRM bisher nicht in der Lage ist eine dichte Interpretation einer Prämisse (siehe Abb. 1 in Kapitel 2.1) zu behandeln, ist eine Erweiterung auf eine dichte Handhabung der Prämissen unablässig. Bisher wird bei der Überprüfung einer Prämisse nur das relative Objekt in Richtung seines Ankers (lokalisierendes Objekt) verschoben und dann geprüft, ob sich eine Verletzung der Prämisse ergibt. Diese Verschiebung findet bisher nur entlang einer Relation statt. Eine Verschiebung innerhalb der Bereiche zwischen den Achsen führt sofort zu einem Gegenbeispiel. Durch die Ergebnisse der Experimente wird ersichtlich, dass auch diese Bereiche zulässig sind und daher von der SRM mit beachtet werden müssen.

Um die Problematik besser zu erläutern, werden zwei Ansätze anhand eines Beispiels erklärt. Es sei folgendes Modell durch Prämissen gegeben.

A ist links von B.
B ist links von C.
D ist über C.

```

<relations>
<relation>
  <name>left </name>
  <abbreviation>l</abbreviation>
  <base>>true</base>
  <arity>2</arity>
  <domain>spatial</domain>
  <opposite>right</opposite>
  <transitivity>transitive</transitivity>
  <reflexivity>non-reflexive</reflexivity>
  <symmetry>non-symmetric</symmetry>
  <range>
    <min>1</min>
    <max>-1</max>
    <step>1</step>
  </range>
  <semantic>
    <conj>
      <term>
        <arg>1</arg>
        <arg>2</arg>
        <rel>left</rel>
      </term>
    </conj>
  </semantic>
  <varOrder>
    <direction>B</direction>
    <direction>F</direction>
  </varOrder>
</relation>
<relation>
  ...
</relation>
</relations>

```

Abbildung 5: Beispiel einer XML Datei mit deren Hilfe eine Relation beschrieben werden kann. Hier die Relation *links*. Diese Form der Darstellung ist noch experimentell, Details dazu siehe Studienarbeit “Modellbasierte Verarbeitung allgemeiner räumlicher Relationen” von Felix Steffenhagen.

Obleich das Problem keine Negation enthält, kann man an der Position des Objektes D über dem Objekt C gut erkennen, wie sich ein Problem ergibt, wenn man nach der Konklusion “gilt immer? A ist links von D” fragt. Im bisherigen Ansatz würde bereits eine erste Überprüfung die Prämisse als verletzt abweisen und das initiale Modell wäre bereits ein Gegenbeispiel. Korrekt wäre aber das Objekt D so lange zu verschieben, bis es sich über dem Objekt A befindet. Die Prämisse “D ist über C” würde in keinem Fall verletzt werden, jedoch die gefragte Konklusion “A ist links von D”. Im ersten Experiment gaben die Versuchspersonen eine eindeutige Grenze an, wann eine Prämisse noch gilt und wann nicht mehr.

Zur Lösung dieses Problems gibt es primär zwei Möglichkeiten. Bei der ersten Methode kann man den Fokus jedes Mal durch das Modell laufen lassen und dabei prüfen, ob die Prämissen verletzt wurden. Ausgehend vom oberen Beispiel sähen die einzelnen Schritte wie folgt aus: Im ersten Schritt würde der Fokus das Objekt D suchen und

dann über das Objekt C zum Objekt B und schließlich zum Objekt A laufen. Die dabei durchgeführten Fokusoperationen werden gespeichert und schließlich ausgewertet. Die Prämissen würden nicht verletzt und damit ein neuer Variationsschritt durchgeführt. Dabei wird das Objekt B um eine Position nach links verschoben und anschließend die Konklusion geprüft. Der Fokus bewegt sich zum Objekt D von dort zu dem Objekt B und schließlich zu dem Objekt A. Obwohl sich das Objekt D nun nicht mehr über dem Objekt C befindet ist die Prämisse, dass sich das Objekt D immer noch im Bereich über dem Objekt C befindet, nicht verletzt. Auch die Konklusion die abgefragt wird, ist nicht verletzt. Im nächsten Schritt wird das Objekt D nun um eine weitere Position nach links verschoben. Der Fokus sucht das Objekt D und geht direkt zu dem Objekt A über. An diesem Punkt ist die Konklusion "A ist links von D" verletzt und das Modell wird als Gegenbeispiel zurückgegeben.

Die zweite Methode greift bei der Inspektion auf die Koordinaten des Systems zurück. Im ersten Augenblick mag diese Methode als nicht dem Menschen nachempfunden gelten, aber ein Problem der SRM gegenüber dem Menschen ist, dass ein Mensch in der Lage ist, Objekte anhand ihrer relativen Positionen zu analysieren. Im Fall der SRM muss hierbei auf eine Hilfskonstruktion, in diesem Fall Koordinaten, zurückgegriffen werden. Wieder am oberen Beispiel orientiert, sähe die Variation und Validation folgendermaßen aus: Der Fokus würde das Objekt D suchen. Die Position des Objektes A gilt als Ausgangsposition und wird mit $(0,0)$ gespeichert. Nachdem das Objekt D gefunden wurde, stehen auch die Koordinaten relativ zum Objekt A fest: zwei Schritte nach rechts in X-Richtung und ein Schritt nach oben in Y-Richtung. Die sich daraus ergebenden Koordinaten wären dann $(2,1)$. Nach der Definition des kompletten Systems aus Kapitel 2.2 ist damit die Prämisse nicht verletzt. Auf die gleiche Weise werden nun auch alle anderen Prämissen geprüft. Gibt es keine Verletzung, wird der nächste Variationsschritt durchgeführt. Nach dem Verschieben des Objektes D um eine Position nach links hat Objekt D die Koordinaten $(1,1)$. Nach der Definition des kompletten Systems liegt keine Verletzung vor. Im nächsten Schritt verschiebt sich das Objekt D nun über die Position des Objektes A und hat damit die Koordinaten $(0,1)$. Das verletzt die Definition der Prämisse *links*. Damit ist ein Gegenbeispiel gefunden. Um bei dieser Methode den Fokus nicht an Hand der Koordinaten durch den Layer zu bewegen, muss noch ein Weg gefunden werden dem Fokus einen Pfad zu übergeben, wie er sich durch das Modell bewegen soll. Dieses Problem lässt sich mit einem Graphen lösen. Dabei wird aus dem Modell ein Graph erzeugt und über diesem der kürzeste Weg von Objekt A zu Objekt B bestimmt. Dieser kann beispielsweise mit dem Algorithmus von Dijkstra erzeugt werden [Dij59]. Ein Punkt der hierbei beachtet werden sollte, ist die Art, wie der Graph erzeugt werden soll. Zwei Ansätze kommen hier in Frage. Der erste betrachtet nur Kanten zwischen direkt benachbarten Objekten. Das bedeutet, nur Objekte die direkt nebeneinander liegen, werden durch eine Kante verbunden. Der zweite Ansatz bestimmt, dass alle Objekte mit einer Kante verbunden sind, die im Verständnis einer Relation nebeneinander liegen. Das bedeutet, ausgehend von einem Objekt wird geprüft, welches Objekte links, rechts, über oder unter diesem liegen. Das beinhaltet auch Objekte die ein oder mehrere leere Positionen vom Ursprungsobjekt entfernt liegen. Die Entfernung wird als Gewichtung der Kante gespeichert. In Abb 6

sind die Unterschiede der Ergebnisse dargestellt.

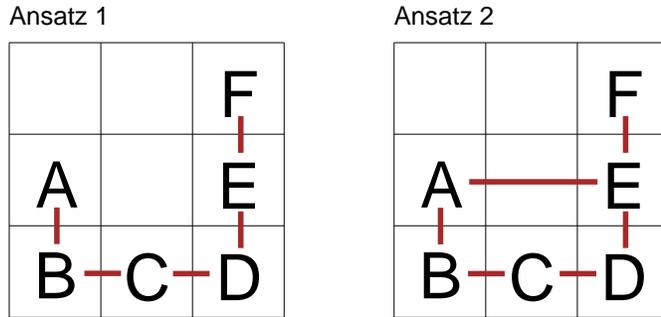


Abbildung 6: Unterschiede der beiden Ansätze

Der erste Ansatz bildet die Kanten nur zwischen den benachbarten Objekten. Wenn der Fokus sich von A nach F bewegen soll, muss er dabei den kompletten Weg durch das Modell zurücklegen. Beim zweiten Ansatz kann der Fokus sich direkt von A nach E bewegen und dann zu F. Die Kosten würden in diesem Fall jedoch nicht “1” betragen, da die Kante zwischen A und E über zwei Positionen verläuft.

Zuletzt bleibt noch die Variationsphase. Während der Variation wurde ein Objekt bisher entlang einer Relation zu seinem Anker verschoben. Objekte die dabei zwischen dem variierten Objekt und dem Ankerobjekt liegen, werden dabei mit dem variierten Objekt vertauscht. Der Anker ist das Objekt, welches bei der Konstruktion als RO der Prämisse steht. Im Fall der Prämisse, “A ist links von B” und “A ist links von C”, wäre A das Ankerobjekt von C. B hat in diesem Beispiel keinen Anker, da es als erstes verarbeitet wird und deshalb kein Objekt die direkte Position links neben dem A belegt. Wenn ein Objekt, wie im oberen Beispiel das C, keine eindeutige Position hat, wird es mit seinem Anker und der Relation zu diesem Annotiert. Wenn ein Objekt aus einer negierten Prämisse in das Modell eingefügt wird, wird dieses ebenfalls Annotiert. In diesem Fall beinhaltet die Annotation die negierte Relation und das Ankerobjekt. In den Experimenten zeigte sich, dass Menschen im Fall einer negierten Prämisse auch Variation in einer anderen Dimension betrachten. Deshalb muss die Variation in der Lage sein, das Objekt auch innerhalb der anderen Dimensionen zu variieren. Dazu ergeben sich drei mögliche Herangehensweisen. Die erste Möglichkeit besteht darin, das annotierte Objekt zunächst wie gehabt zuerst entlang der Relation zu seinem Anker zu bewegen. Sobald der Anker erreicht ist, wird die Dimension die orthogonal dazu liegt abgearbeitet. Da diese Dimension zwei Relationen besitzt wird zunächst nur eine der Relationen geprüft. Dazu wird das Objekt an alle Positionen die sich zwischen Anker und der ursprünglichen Position und der zu prüfenden Relation befinden variiert. Nach Erreichen der Position wird die verbleibende Relation durch variieren abgearbeitet. In Abb. 7, I) ist die Reihenfolge zu sehen. Die zweite Möglichkeit besteht darin, zuerst entlang der annotierten Relation zu variieren dann jedoch die beiden Relationen der orthogonalen Relation im Wechsel zu verarbeiten, siehe Abb. 7, II). In der dritten

Möglichkeit werden, wie in der zweiten, die Relationen der orthogonalen Dimensionen im Wechsel verarbeitet. Anders als bei den anderen beiden Ansätzen wird die orthogonale Dimension bereits bei der Bewegung des Objektes hin zum Anker vollzogen. (Siehe Abb. 7, III). Eine Variation aller drei Möglichkeiten wäre, während die orthogonalen Relationen verarbeitet werden, ebenfalls alle Positionen, auch diese zwischen den Objekten, zu betrachten.

4.2 Konkrete Implementierung

4.2.1 Grundlegende Änderungen

Die ersten Veränderungen wurden in der Konstruktion vorgenommen. Zuerst wurde festgelegt, wie eine negierte Relation verarbeitet werden soll. Dazu wurde der Input der SRM erweitert. Bisher wurden die Prämissen als Tripel entweder per Textdatei oder durch die Eingabemaske des Programms übergeben. Die Formatierung bestand dabei aus der Relation, als erste Position, und den Argumenten an zweiter und dritter Position. Bei der Erweiterung wurde diese Struktur beibehalten und die Negation an die erste Stelle gesetzt. Die Negation wird als Boolean übergeben, mit den Werten *True* oder *False* als Möglichkeit. Daraus ergibt sich nun ein Quadrupel der Form (Negation, Relation, Argument1, Argument2). Alle Funktionen, die bisher das Tripel verwendet haben, wurden dahin gehend verändert, dass sie ein Quadrupel verarbeiten können. Im Einzelnen wurden, der Prämissenparser, die Funktion zum Auslesen der Prämissen, die Konstruktionsphase und die Inspektion angepasst.

4.2.2 Konstruktion

Die Konstruktionfunktion ist dafür zuständig, die Prämissen zu verarbeiten und zu interpretieren. Die Objekte aus den Prämissen werden dann in einen Layer eingefügt, der aus einem zweidimensionalen Array besteht. Die Objekte werden anhand von Koordinaten eingefügt, wobei die X- und Y-Werte der Adressierung im Array entsprechen. In Python eignet sich dafür die Datenstruktur *dictionary*. Die Änderungen in der Konstruktion sind im Vergleich zur Inspektion und Variation relativ gering ausgefallen. Die Wichtigste Veränderung ist die Interpretation einer negierten Prämisse. Wenn eine negierte Prämisse von der Konstruktion verarbeitet wird, muss zuerst die konverse Relation gebildet werden. Dies geschieht durch folgende neue Funktion.

Die Funktion prüft zuerst, ob eine Prämisse genau zwei Argumente enthält. Dies ist wichtig, da Negation nur für binäre Relationen definiert werden kann. Im zweiten Teil wird die Prämisse konvertiert. Hierbei bleibt die Reihenfolge der Argumente erhalten und die Relation wird durch die entgegengesetzte Relation ersetzt. Was genau diese Relation ist, wird durch die Beschreibung der Relation in der XML-Datei festgelegt. Die Zeile darunter kopiert das *range*-Attribut des Objektes. Zum Schluss wird das *Negations*-Attribut kopiert. Dieses wird für die Annotation benötigt. Die Prüfung findet in der *construct()* Methode statt, siehe Abb. 9.

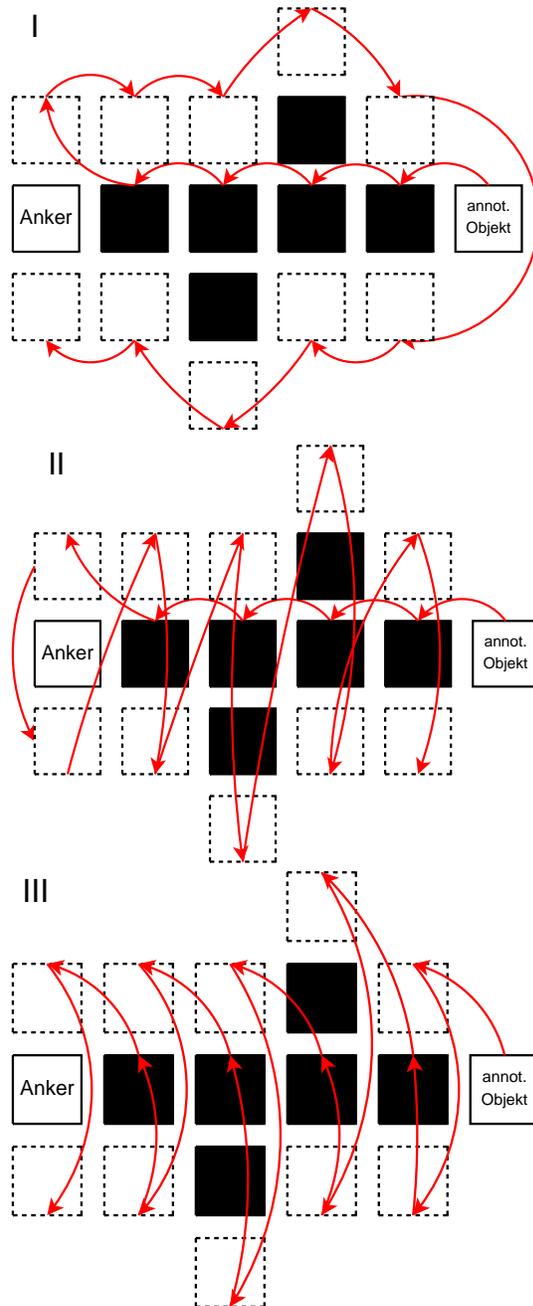


Abbildung 7: Verschiedene Variationsbewegungen. Die schwarzen Bereiche markieren Objekte innerhalb des Modelles, die gepunkteten möglichen Positionen für das annotierte Objekt. Die roten Pfeile zeigen die Bewegung des varierten Objektes, ausgehend von der Ausgangsposition des Objektes.

```

def converse():
    if Anzahl der Argumente > 2
        return
    if entgegengesetzte Relation existiert
        behalte Reihenfolge der Argumente
        tausche Relation mit entgegengesetzter Relation
return neue Prämisse

```

Abbildung 8: Die Struktur der Funktion `converse()`, definiert in der Klasse `Premise`.

```

if Prämisse negiert
    Prämisse = konverse Prämisse
    ...
# Prämissen vom Typ 1
if keines der beiden Objekte im Layer
    Neuer Layer = Layer(Annotations, Kosten)
    füge beide Objekte in neuen Layer ein
# Prämissen vom Typ 2
elif ein Objekt bereits im Layer
    füge fehlendes Objekt in Layer ein
    annotiere falls notwendig
# Prämissen vom Typ 3
elif beide Objekte bereits im Layer
    if Layer nicht identisch
        verschmelze beide Layer
    else
        if neue Prämisse hält im Modell
            return
        else
            try
                korregiere Modell
            except
                Fehler!

```

Abbildung 9: Ausschnitt aus der Struktur der `construct()` Methode.

Die neu erzeugte Prämisse wird verwendet, um die Objekte im Layer einzufügen. Hierbei wird unterschieden ob keines, eines oder bereits beide Objekte im Layer eingefügt wurden. Sind beide Objekte bereits im Layer, wird geprüft, ob die neue Prämisse zu Widersprüchen führt. Ist eines der beiden Objekte bereits im Layer, unterscheidet die SRM zwischen zwei Fällen. Im ersten Fall ist das RO bereits im Layer. Das RO ist das Objekt, das an erster Stelle der Prämisse steht, also das Objekt, welches relativ zu dem anderen Objekt steht. Dann wird die Prämisse unverändert verwendet und das LO entsprechend der Relation eingefügt. Das LO (lokalisierende Objekt) ist das Objekt, welches als zweites in der Prämisse vorkommt. Ist das LO bereits im Layer wird die Prämisse umgedreht. Das bedeutet, die Relation wird durch die konverse Relation ersetzt, die Anordnung der Argumente vertauscht und das RO eingefügt. Sollte keines der beiden Objekte im Layer liegen, so wird ein neuer Layer erzeugt, in den die Objekte entsprechend der Relation eingefügt werden. Für den Fall, dass ein Objekt aus diesem neuen Layer mit einem Objekt des alten Layers in Relation gesetzt wird,

können beide Layer verschmolzen werden, vorausgesetzt das Modell bleibt auch dann konsistent.

4.2.3 Inspektion

Bei der Inspektion wurde die gesamte Prüfroutine der SRM neu geschrieben. Der Grund lag, wie eingangs bereits erwähnt, in der Tatsache, dass die SRM nicht auf einer kompletten Betrachtung des Raums interpretiert. Die bisherige Methode versuchte einfach, ausgehend vom LO der Konklusion, entlang der Relation das RO zu finden. Die neue Methode setzt an einem anderen Punkt an. Zuerst werden dazu die Koordinaten beider Objekte bestimmt, dies geschieht durch die Funktion *getCoordsByName()*, siehe Abb. 10.

```
def getCoordsByName()
    for o in der Menge der Objekte
        if o.name = gesuchtes Objekt
            return Position des Objektes
    return None
```

Abbildung 10: Struktur der getCoordsByName() Funktion

Sobald die Koordinaten der Objekte bestimmt wurden, werden diese anhand einer der folgenden vier Funktionen, siehe Abb. 11 verglichen. Welche Relation geprüft werden muss, wird direkt aus der Konklusion ausgelesen, siehe Abb. 12.

```
def isLeft(X, Y)
    if (x') < (y')
        return True
    else
        return False

def isRight(X, Y)
    if (x') > (y')
        return True
    else
        return False

def isFront(X, Y)
    if (x''') < (y''')
        return True
    else
        return False

def isBehind(X, Y)
    if (x''') > (y''')
        return True
    else
        return False
```

Abbildung 11: Die vier Funktionen zum Testen der Relation aus der Konklusion

```

    if relation = links
        ret = isLeft(LO, RO)
    elif relation = rechts
        ret = isRight(LO, RO)
    elif relation = unter
        ret = isFront(LO, RO)
    elif relation = über
        ret = isBehind(LO, RO)
    if premise.isNegated():
        ret = not ret
    return ret

```

Abbildung 12: veränderte Prüfroutine der Inspektion

Sobald die Konklusion erfüllt ist, liefert die Funktion *True* zurück, wenn die Konklusion nicht erfüllt ist, respektive *False*. Der genaue Code, der für die Erzeugung der Kanten und das Erstellen des Graphen zuständig ist, ist in Abb. 13 dargestellt.

```

def buildGraph()
    for o in Menge aller Objekte
        leftobjects = alle Objekte die sich links befinden
        try:
            leftobject = nächstgelegenes Objekt aus der Menge leftobjects
            dist = Distanz der Objekte zueinander
        except:
            pass
        rightobjects = alle Objekte die sich rechts befinden
        try:
            rightobject = nächstgelegenes Objekt aus der Menge rightobjects
            dist = Distanz der Objekte zueinander
        except:
            pass
        frontobjects = alle Objekte die sich rechts befinden
        try:
            frontobject = nächstgelegenes Objekt aus der Menge frontobjects
            dist = Distanz der Objekte zueinander
        except:
            pass
        behindobjects = alle Objekte die sich rechts befinden
        try:
            behindobject = nächstgelegenes Objekt aus der Menge behindobjects
            dist = Distanz der Objekte zueinander
        except:
            pass
    return graph

```

Abbildung 13: *buildGraph()* Methode, zur Bestimmung der direkten Nachbarn und deren Distanz zum Objekt.

Diese Prüfung reicht jedoch nicht, da in diesem Schritt noch keine Kosten bestimmt werden. Um die Kosten für die Vergleichsoperation zu bestimmen, wird der Fokus vom LO zum RO versetzt. Die dabei entstehenden Kosten sind für den Vergleich notwendig. Damit der Fokus auch einen realistischen Weg zurücklegt, muss zunächst der Pfad bestimmt werden. Dies geschieht mit Hilfe des Algorithmus von Dijkstra [Dij59]. Die

Konstruktion des Graphen geschieht innerhalb der SRM über eine Menge von Kanten, die aus dem Layer erzeugt werden. Der Layer besteht aus einem *dictionary*, in dem die Objekte des Layers mit Pointern über die Koordinaten abgelegt sind. Um den Graphen zu erzeugen, wird ausgehend von jedem Objekt in allen vier Relationsrichtungen nach benachbarten Objekten gesucht. Dazu werden alle Objekte die in den jeweiligen Richtungen liegen ausgegeben und das nächstgelegene gewählt. Zuletzt muss noch die Entfernung zu diesem Objekt bestimmt werden. Dazu wird der Betrag der Differenz der X- und Y-Koordinate des LO und des RO bestimmt und übergeben. Das so entstandene Tripel hat die Form (LO, RO, Entfernung), wobei das LO das Objekt ist, von dem ausgegangen wird. Sobald alle Objekte abgearbeitet sind, wird die Menge der Kanten an die *srn_graph.py* übergeben und der Graph erzeugt. Die Menge der Knoten wird dabei dynamisch aus den Objekten erzeugt, die aus den Kanten hervorgehen. Nachdem der Graph erzeugt und der kürzeste Pfad ermittelt ist, wird eine Liste von Knoten zurückgegeben. Der Fokus bewegt sich dann innerhalb der Liste von einem Objekt zum nächsten und kann so die korrekten Kosten für die Inspektion bestimmen. Im Fall, dass die Prüfung *False* liefert, kann keine Fokusbewegung vom LO zum RO durchgeführt werden. In diesem Fall müssen dennoch Kosten bestimmt werden können. Tritt dieser Fall ein, werden die Kosten über die Zahl der Objekte bestimmt, welche sich in Richtung der Relation zum LO befinden. Befinden sich beispielsweise 3 Objekte in dieser Richtung, betragen diese Kosten 3.

4.2.4 Variation

Der bisherige Code für die Variation mußte im Zuge dieser Arbeit komplett neu geschrieben werden, da die bisherige Variationsmethode durch die Änderungen der Inspektion und der Prämissenrepräsentation nicht kompatibel war. Zusätzlich wurde eine Möglichkeit implementiert, einen Chunk zu bilden. Ein Chunk ist eine Gruppierung von zwei oder mehreren Objekten aus einem Modell. Die Objekte werden dazu in eine neue Liste transferiert und aus dem eigentlichen Modell entfernt. Anstatt der entfernten Objekte wird ein Substitutionsobjekt benutzt. Als Substitutionsobjekt wird in der aktuellen Version das zu variierende Objekt verwendet und ausgehend von diesem die Liste der Chunkobjekte bestimmt. Das Substitutionsobjekt wird für die Variation benötigt, um die Objekte des Chunks sinnvoll variieren zu können. Die Variation mit einem Chunk ist formal korrekt, da ein Objekt nur eine Annotation besitzt. Ein annotiertes Objekt besitzt also genau einen Anker. Objekte, die dieses Objekt als Anker besitzen, können keinen anderen Anker besitzen. Diese rekursiv betrachteten Objekte können folglich keinen Anker außerhalb der Kette besitzen. Wenn man das Modell in einen Baum konvertieren würde, so wäre das Ausgangsobjekt die Wurzel eines echten Teilbaums. Alle Objekte, die einen Anker haben, der Teil dieses Baums ist, sind, da es nur einfach annotierte Objekte gibt, wieder Teile dieses Teilbaums. Wenn man diesen Teilbaum auf seine Wurzel reduziert und diese im Baum verschiebt und wieder expandiert, bleiben alle Relationen innerhalb des Teilbaums unverändert. Der Chunking-Ansatz, der in die SRM implementiert wurde, funktioniert genau nach dieser Methode, die bei normalen und bei negierten Annotationen angewendet wird.

4 Implementierung

Bei einer negierten Annotation ist es unbedingt notwendig Chunking zu verwenden, da bei der Variation die Dimension verlassen wird und eventuell annotierte Objekte mit variiert werden müssen.

Um bei der Variation eine Konklusion zu testen, wird versucht, eines der beiden Objekte über das andere hinwegzutauschen, um ein Gegenbeispiel zur gegebenen Konklusion zu erzeugen. Dazu muss zunächst geprüft werden, welche Objekte der Konklusion annotiert sind. Ist keines annotiert, muss lediglich die Konklusion mittels der *Inspect()* Methode geprüft werden. Wird die Konklusion verletzt, so ist das Ausgangsmodell bereits das gesuchte Gegenbeispiel. Andernfalls gilt die Konklusion in allen Modellen. Ist nur eines der Objekte annotiert, wird versucht dieses in Richtung des anderen zu variieren. Bevor ein Objekt variiert werden kann, wird mittels der Annotationen geprüft, ob dieses bereits Anker anderer Objekte ist. Diese Prüfung wird rekursiv durchgeführt. Gibt es solche Objekte, werden diese zuerst variiert. Auf diese Weise werden alle Objekte, die von dem Objekt aus der Konklusion abhängig sind, gruppiert, um einen Chunk zu bilden. Die Struktur der Methode ist in Abb. 14 zu sehen.

```
def buildChunk():
    Obj.Chunk = [ Objekte mit Annotation die Objekt als Anker haben ]
    if not Obj.Chunk:
        keinen Chunk bilden
    while True:
        for X in Obj.Chunk:
            annots = Annotations(X)
            Neue.Obj = [ Objekte die an erster Stelle in annots stehen ]
            Obj.Chunk.extend( füge Objekte Chunk hinzu )
        if count == len(Obj.Chunk):
            beende, da keine neuen Objekte mehr hinzukommen
    for O in Obj.Chunk:
        entferne Objekte aus Liste Obj.Chunk aus Modell
```

Abbildung 14: Struktur der *buildChunk()* Methode

Sind beide Objekte der Konklusion annotiert, wird zunächst nur eines der beiden Objekte variiert bis entweder keine Variation mehr möglich ist oder ein Gegenbeispiel erzeugt wurde. Erst danach wird das zweite Objekt variiert. Wird auch bei dieser Variation kein Gegenbeispiel gefunden, so gilt die Konklusion in allen möglichen Modellen und ist somit zulässig. Die Prüfung der Konklusion und die Variation der Objekte wird mittels der *checkConlusion()* Methode durchgeführt, siehe Abb. 15.

Wie in der Abbildung zu sehen ist, wird das variierte Objekt schrittweise zum Anker bzw. dem Konklusionsobjekt bewegt. Dies hängt davon ab, ob Anker und Konklusionsobjekt in der selben Relationsrichtung liegen oder nicht. Diese Methoden dienen bisher jedoch nur der Prüfung und dem Aufrufen der Variation. Die Variation selber findet durch die Methode *exchange()* bzw. der überwiegend identischen Methode *exchange_neg()* statt. Wie die Namen bereits erkennen lassen, werden negierte Annotationen in einer separaten Methode abgearbeitet. Der Grund dafür ist, die *exchange()* Methode muss Annotationen ohne Negation wie bisher in der selben Dimension variieren. Handelt es sich jedoch um eine negierte Annotation, so wird innerhalb der

```

def checkConclusion():
    if not Konklusion gilt in Modell:
        return False
    if Konklusion or inverse Konklusion is Annotation:
        return True

    try:
        variiere RO Richtung LO
    except CounterExampleFoundException:
        return False

    try:
        variiere LO Richtung RO
    except CounterExampleFoundException:
        return False
    return True

```

Abbildung 15: Struktur der *checkConclusion()* Methode

exchange() Methode *exchange_neg()* aufgerufen. Von den in Abbildung 7 beschriebenen Möglichkeiten, wie die Variation im Fall einer negierten Annotation vollzogen werden soll, wird die letzte verwendet. Die *exchange_neg()* geht dabei wie folgt vor. Zuerst wird die Richtung ermittelt, in die variiert werden soll. Wurde die Richtung bestimmt, wird aus der Relation eine Variationsordnung ausgelesen. Wie in Abbildung 5 zu sehen ist, befinden sich unter dem Knoten `<varOrder>` zwei Relationen. Diese werden in einer Liste zwischengespeichert. Innerhalb der *exchange_neg()* Methode bewegt sich der Fokus einen Schritt in Richtung des Zielobjektes. Anschließend bewegt sich der Fokus in Richtung der ersten Relation aus `<varOrder>` bis zur ersten freien Position. Das Variationsobjekt wird dort platziert und ein Inspektionsschritt durchgeführt. Danach wird dieses Objekt wieder an die Ausgangsposition verschoben. Von dort bewegt sich der Fokus wieder einen Schritt in Richtung des Zielobjektes, dann jedoch in Richtung der zweiten Relation aus `<varOrder>` bis zur ersten freien Position. An dieser Position wird das Objekt eingefügt und eine Inspektion gestartet. Nachdem das Objekt erneut an die Ausgangsposition verschoben wurde, wird es, wie in der *exchange()* Methode, in Richtung des Zielobjektes variiert. Diese Bewegungen werden so lange wiederholt bis entweder ein Gegenbeispiel erzeugt wurde, oder keine weiteren Variationsschritte mehr möglich sind. Die Strukturen beider Methoden sind in Abbildung 16 zu sehen.

```

def exchange():
    if Annotation ist negiert:
        exchange_neg()
    target = Zielobjekt aus Konklusion
    target_dirctn = Richtung des Zielobjekties
    anchor = bestimme Anker

    berechne Chunk

    while not Anker erreicht:
        neighbour = Nachbar in target_dirctn
        if not neighbour:
            resubstituiere Chunk
            return
        variiere Chunk in target_dirctn
        if Konklusion verletzt
            resubstituiere Chunk
            raise Gegenbeispiel gefunden

    if Anker erreicht:
        resubstituiere Chunk
        if Anker ist annotiert
            erzeuge neue Konklusion mit Anker als neuem Objekt
            rufe exchange() mit neuer Konklusion auf

def exchange_neg()
    ...

    for Relationen aus varOrder:
        bewege Fokus einen Schritt Richtung Zielobjekt
        bewege Fokus fff in Relationsrichtung aus varOrder

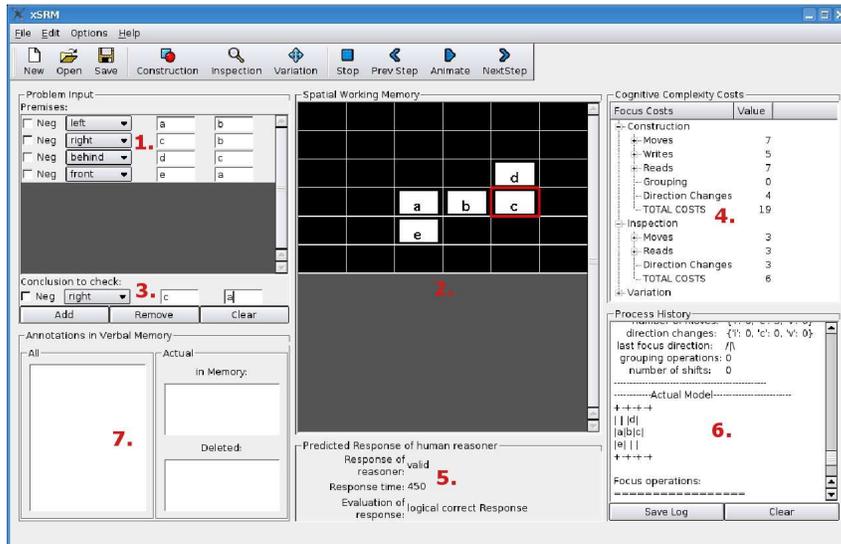
        if Konklusion verletzt:
            resubstituiere Chunk
            raise Gegenbeispiel gefunden
        bewege Fokus an Ausgangsposition

    ...

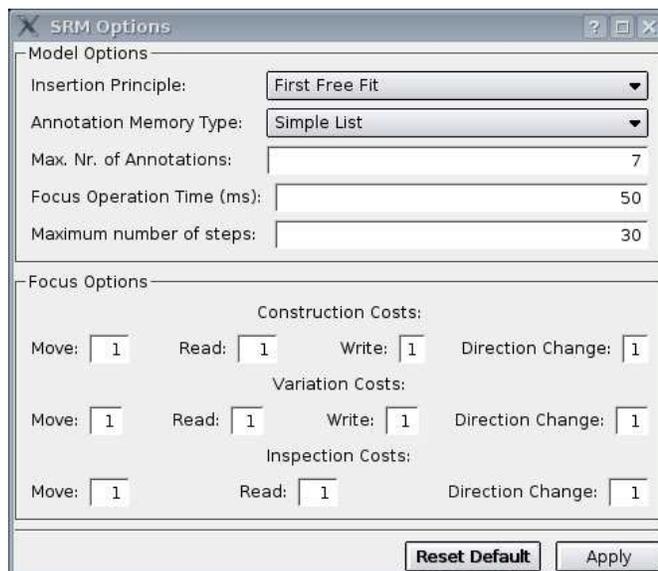
```

Abbildung 16: Strukturen der *exchange()* und *exchange_neg()* Methoden

4.3 Screenshots

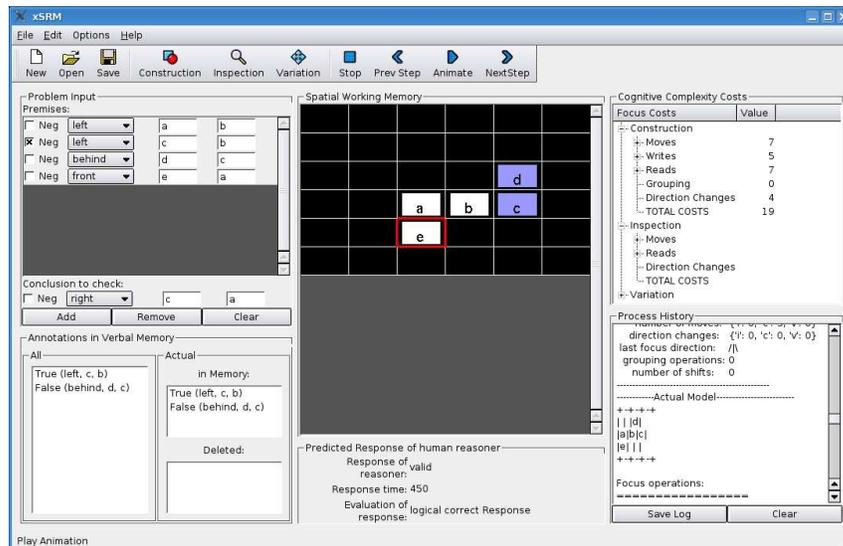


Screenshot der SRM. 1: Prämissen Eingabefeld. 2: Bereich für die Darstellung des Modells. 3: Eingabe für die zu prüfende Konklusion. 4: Anzeige der Kosten. 5: Anzeige für das Resultat der Inspektion. 6: Fenster für Logfile. 7: Fenster in dem Annotationen angezeigt werden.

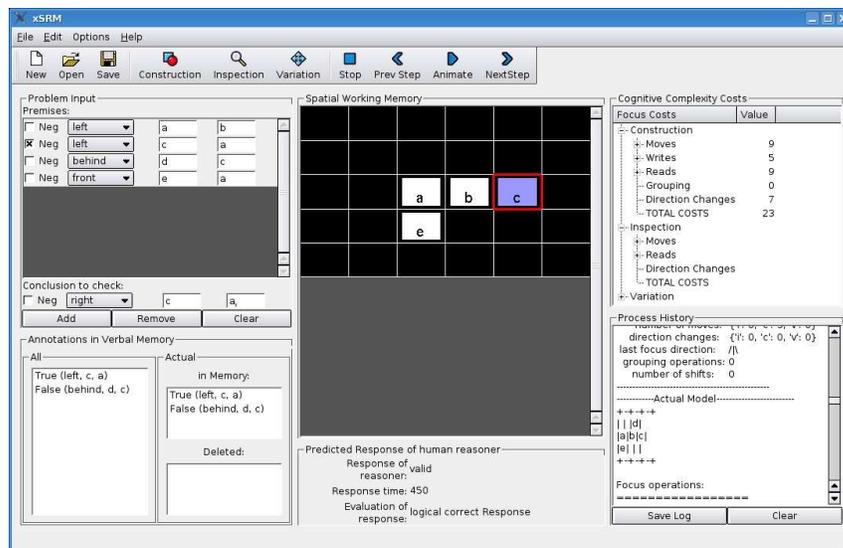


Optionsfenster der SRM.

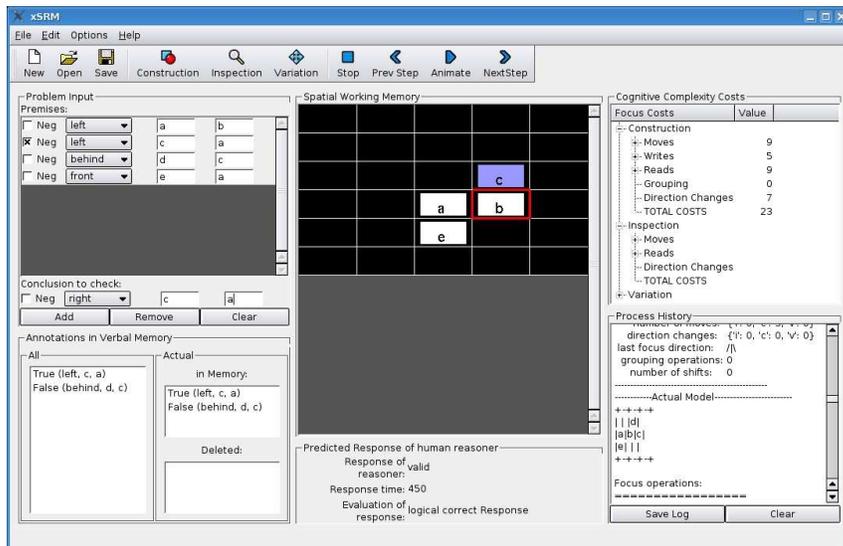
4 Implementierung



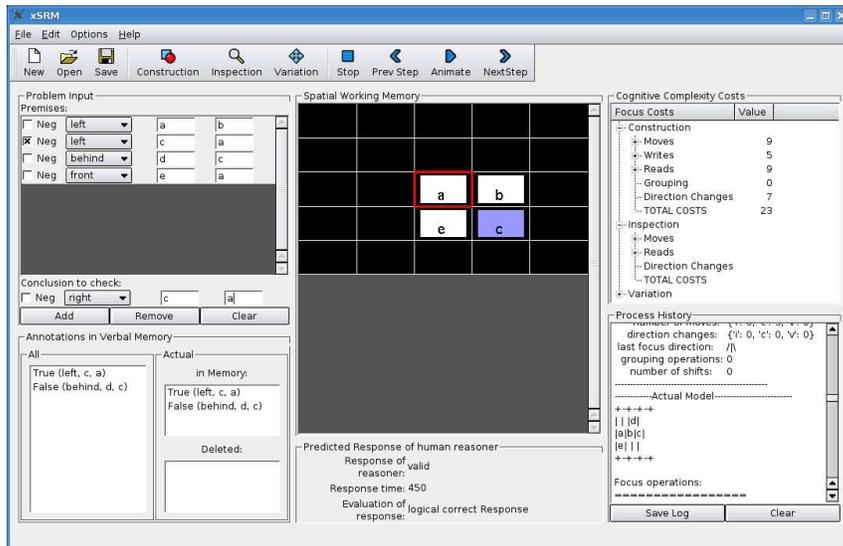
Eingabe eines Modells mit negierter Prämisse. Das betreffende Objekt "C" wurde annotiert (violett eingefärbt), ebenso das Objekt "D", das "C" als Ankerobjekt hat.



Chunking: Beim Prüfen der Konklusion "C rechts A" wird "D" in den Chunk aufgenommen. Das verbleibende Objekt "C" dient als Repräsentation für den Chunk.



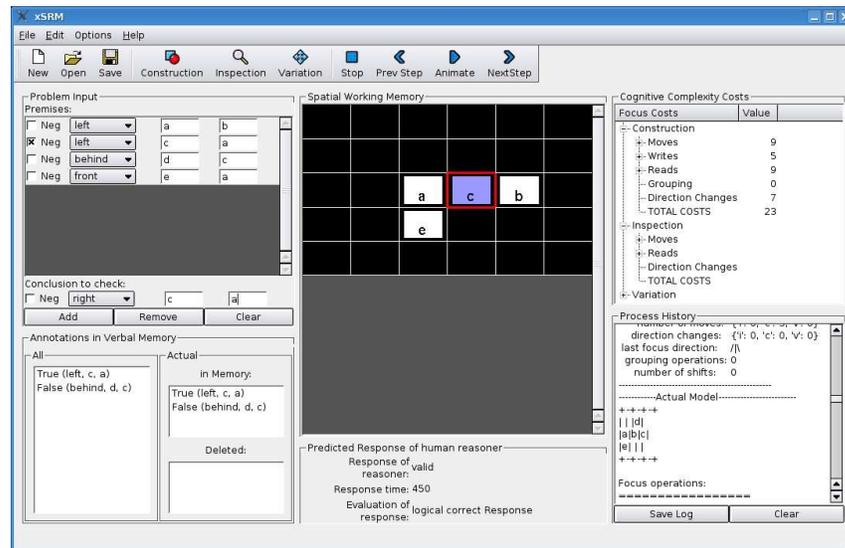
1. Schritt



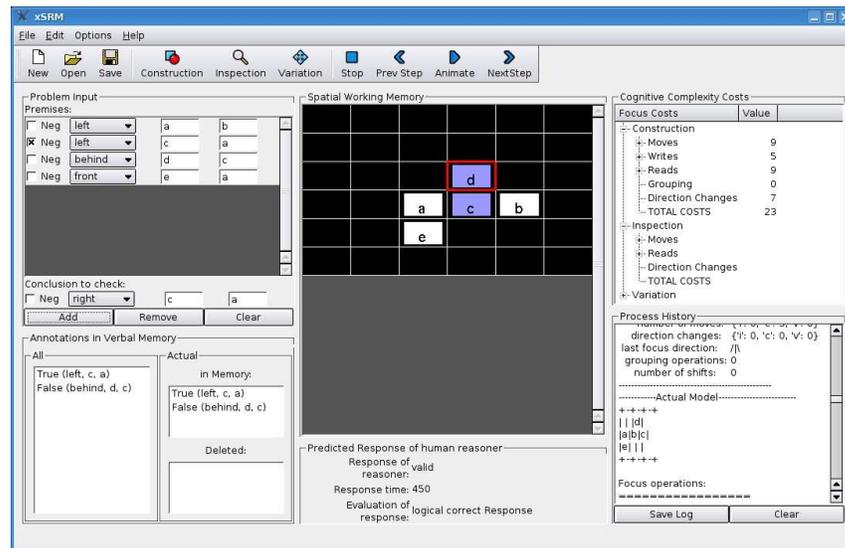
2. Schritt

Zwischenschritte der Variation. Der Chunk wird zunächst oberhalb von "B" platziert. Im nächsten Schritt wird der Chunk unterhalb platziert. In jedem Schritt findet ein Inspektionschritt statt.

4 Implementierung



Vertauschen der beiden Objekte "B" und "C".



Nach Erreichen des Ankers von "C" wird ein Resubstitutionsschritt durchgeführt. Dazu wird das vorher im Chunk befindliche "D" wieder in das Modell eingefügt.

5 Ergebnisse

5.1 Experimente

Die Ergebnisse der Experimente haben bezüglich Negation beim räumlichen Schließen gezeigt, dass die theoretischen und auch formalen Überlegungen korrekt und anwendbar sind. Während der Experimente hat sich eine klare Tendenz hin zu der Vermutung gezeigt, dass eine negierte Prämisse dadurch behalten wird, indem die konverse Relation während der Konstruktion benutzt wird und eine Annotation mit der originalen (negierten) Prämisse gemerkt wird. Diese Methode scheint den besten Kompromiss zwischen 'ökonomischem' Speichern einer Information und dem Wiedergeben möglichst vieler Variationen zu sein. Bei der Auswertung des ersten Experiments hat sich ebenso gezeigt, dass eine dichte Interpretation des Raums dem entspricht, was Menschen beim Schließen mit mentalen Modellen verwenden. Die Auswertungen des zweiten Experiments konnten dies bestätigen. Hier wurden Modelle konstruiert, die den Schluss zulassen, dass diese Art der Interpretation als wahrscheinlich angenommen werden kann. Diese Varianten waren nicht signifikant stabil in Bezug auf das präferierte Modell. Jedoch zeigt dies, dass die Versuchspersonen sich darüber im Klaren sind, welche Variationsmöglichkeiten mit der Negation der Prämissen einhergehen. In einem Teil des zweiten Experiments, der nicht in die Auswertung eingegangen ist, konnte diese Tatsache gut beobachtet werden. Die Versuchspersonen waren angehalten alle ihrer Meinung nach möglichen Modelle zu zeichnen, die sich aus den Prämissen ergeben. Um eine Reihenfolge der Präferenzen ermitteln zu können, sollten die gezeichneten Modelle nummeriert werden. Eine erste Analyse der Aufgaben zeigte, dass ausnahmslos alle Teilnehmer das in Frage kommende Objekt in allen Dimensionen positionierten. Ebenso wurden auch die Positionen gewählt, die nicht direkt in einer Richtung der Relation lagen, also auch diagonale Positionen betrachtet wurden. Mit Hilfe des dritten Experiments konnte schließlich auch die Frage nach der Komplexität eindeutig geklärt werden. Wie bereits im Vorfeld vermutet wurde, sank die Korrektheit der erzeugten Modelle mit Negation gegenüber in-determinierten Problemen. Diese Tendenz wurde bereits im zweiten Experiment festgestellt, dennoch konnte erst das dritte Experiment statistische Klarheit verschaffen.

5.2 SRM

Ziel der Implementierung in die SRM war, die neu gewonnen Erkenntnisse mit in die Deduktion einzubeziehen, um damit die Deduktionsprozesse des Menschen noch präziser modellieren zu können. Während jeder Instanz der Implementierung wurden mehrere Möglichkeiten der Umsetzung durchgesprochen und gegeneinander aufgewogen, um eine möglichst genaue Annäherung an das menschliche Schließen zu erhalten. Die aktuelle Implementierung verbessert die SRM dahin gehend und durch eine flexible Gestaltung ist es möglich, zu einem späteren Zeitpunkt einfache Erweiterungen einzubauen. Dies ermöglicht andere Verfahren und Theorien zu testen, um eventuelle neue Erkenntnisse zur Verbesserung der SRM zu implementieren. Abschließend bleibt

zu sagen, dass mit der Erweiterung der SRM um Negation ein wichtiger Schritt abgeschlossen wurde, um mit der SRM die menschliche Deduktion besser und präziser zu simulieren.

5.3 Offene Fragen

Die in dieser Arbeit gewonnenen Erkenntnisse tragen dazu bei das menschliche Schließen besser zu verstehen und sollen als Grundlage für das Verständnis der Negation dienen. Dennoch gibt es auf diesem Gebiet noch sehr viel Interessantes, was erforscht werden sollte. Die Beispiele und Aufgaben, die während der Experimente verwendet wurden, enthielten nur eine negierte Prämisse. Was geschieht jedoch im Fall von mehreren negierten Prämissen? In wie weit verändert sich das präferierte mentale Modell, wenn das selbe Objekt durch zwei Prämissen *nicht links* und *nicht rechts* positioniert werden darf? In diesem Fall ist eine wahrscheinliche Voraussage nicht mehr möglich, da eine Positionierung oberhalb oder unterhalb gleich wahrscheinlich erscheinen. Um diese Frage zu beantworten, sind weitere Experimente notwendig.

Ebenso wurden im Zuge dieser Arbeit nur ein- und zweidimensionale Probleme untersucht. Was ist aber mit dreidimensionalen Problemen? Gelten auch hier die Regeln für das Erstellen eines PMM? Vielleicht findet bei dreidimensionalen Problemen eine Art der Vereinfachung statt, bei der das Problem auf einen zweidimensionalen Raum reduziert wird. Die Erkenntnisse der Experimente lassen darauf schließen, dass schon das Schließen mit negierten zweidimensionalen Problemen an der Grenze des Möglichen liegt, was das Potenzial des *visual sketchpad* und des *phonological loop* betrifft. Eventuell könnten Untersuchungen, die gezielt auf die Grenzen dieser Instanzen ausgerichtet sind, diese Frage eindeutig klären.

Ein weiterer Punkt, der zu klären wäre, ist, ob eine andere Fragestellung nicht auch komplett andere Präferenzen mit sich bringt. Während der Experimente waren die Versuchspersonen immer gezwungen ein Modell zu erstellen, was natürlich dazu führt, dass die Information aus der Negation verarbeitet werden muss. Was jedoch passiert wenn man nur danach fragt: Was ist wenn etwas nicht links ist? Bereits durch eine kurze Selbstanalyse wird man feststellen, dass die Frage nicht so einfach zu beantworten ist. Würde man beispielsweise jemand nach dem Weg fragen und die Antwort, “das Gesuchte befindet sich nicht auf der linken Seite” bekommen, wäre damit nicht eindeutig geklärt, ob sich das Gesuchte nun auf der rechten Seite befindet, oder eventuell geradeaus, oder an einer zurück liegenden Stelle.

Ebenso wichtig wie diese eher theoretischen Überlegungen, sind Erweiterungen der SRM. So wäre ein nächster wichtiger Schritt die Veränderung des Verhaltens des Fokus. Dieser ist bisher nicht in der Lage diagonale Bewegungen zu vollziehen. Angeregt durch die Arbeit von Gapp [Gap95] ist jedoch stark anzunehmen, dass diese Art der Bewegung im menschlichen Deduktionsprozess möglich ist. Auch eine Erweiterung um eine weitere Dimension ist vorstellbar. Die meisten Probleme zum räumlichen Schließen des alltäglichen Lebens hängen schließlich mit dem Schließen im dreidimensionalen

zusammen. Diese und andere offenen Fragen sollen dazu anregen weitere Erkenntnisse auf diesem Gebiet zu gewinnen.

Literatur

- [Dij59] DIJKSTRA, EDSEGER. W.: *A note on two problems in connexion with graphs*. Numerische Mathematik, 1:269–271, 1959.
- [Gap95] GAPP, K. P.: *Angle, Distance, Shape and their Relationship to Projective Relations*. In: MOORE, J.D. und J.F. LEHMAN (Herausgeber): *Proceedings of the Seventeenth Annual Conference of the Cognitive Science Society*, Seiten 112–117. Lawrence Erlbaum Associates Inc., 1995.
- [HG06] HASSON, U. und S. GLUCKSBERG: *Does understanding negation entail affirmation? An examination of negated metaphors*. Journal of Pragmatics, 2006.
- [JLB91] JOHNSON-LAIRD, P. N. und R. M. J. BYRNE: *Deduction*. Erlbaum, Hillsdale, NJ, 1991.
- [KLZ06] KAUP, B., J. LUEDTKE und R.A. ZWAAN: *Processing negated sentences with contradictory predicates: Is a door that is open mentally closed?* Journal of Pragmatics, 38:1033–1050, 2006.
- [MJL82] MANI, K. und P. N. JOHNSON-LAIRD: *The mental representation of spatial descriptions*. Mem Cognit, 10(2):181–7, 1982.
- [RFS07] RAGNI, M., T. FANGMEIER und S. SCHLEIPEN: *What about negation in spatial reasoning?* In: *Proceedings of the 28th Annual Cognitive Science Conference (CogSci 07)*. Lawrence Erlbaum Associates Inc., 2007.
- [RFWK06] RAGNI, M., T. FANGMEIER, L. WEBBER und M. KNAUFF: *Complexity in Spatial Reasoning*. In: *Proceedings of the 28th Annual Cognitive Science Conference*, Mahwah, NJ, 2006. Lawrence Erlbaum Associates.
- [RFWKed] RAGNI, M., T. FANGMEIER, L. WEBBER und M. KNAUFF: *Preferred mental models: How and why they are so important in human spatial reasoning*. In: FREKSA, C., M. KNAUFF, B. KRIEG-BRÜCKNER, B. NEBEL und T. BARKOWSKY (Herausgeber): *Spatial Cognition V*. Springer, Berlin, submitted.
- [RHK⁺05] RAUH, R., C. HAGEN, M. KNAUFF, T. KUSS, C. SCHLIEDER und G. STRUBE: *Preferred and Alternative Mental Models In Spatial Reasoning*. Spatial Cognition and Computation, 5:239–269, 2005.
- [RS06] RAGNI, M. und F. STEFFENHAGEN: *An Implementation of the SRM-Model*. Technical report 011-09/2006, SFB/TR 8 Spatial Cognition; <http://www.sfbtr8.uni-bremen.de>, 2006.
- [SRF07] SCHLEIPEN, S., M. RAGNI und T. FANGMEIER: *Negation in Spatial Reasoning*. In: J. HERTZBERG, M. BEETZ, R. ENGLERT (Herausgeber): *Proceedings of the 30th Annual German Conference on AI, KI 2007*, Seiten 175–189. Springer-Verlag Berlin, Heidelberg, 2007.