

Studienarbeit

**Räumliches Schließen über n-stelligen
Relationen: Kognitive Modellierung und
formale Aspekte**

Felix Steffenhagen
steffenh@informatik.uni-freiburg.de
Universität Freiburg

20. Mai 2008

Betreuer:
Prof. Dr. Bernhard Nebel
und Dr. Marco Ragni
{nebel, ragni}@informatik.uni-freiburg.de

Inhaltsverzeichnis

1	Einführung	3
2	Theoretische Grundlagen	6
2.1	Mentale Modelltheorie	6
2.2	SRM	8
2.3	Formale Aspekte des relationalen Schließens	10
2.4	Einfache und komplexe Relationen	12
2.5	Komplexe Relationen in der SRM-Software	13
2.6	Konstruktionsprozess	15
2.7	Inspektionsphase	17
2.8	Variationsprozess	18
3	Implementation	20
3.1	Software-Architektur	20
3.2	Konstruktion	22
3.3	Inspektion	26
3.4	Variation	28
4	Zusammenfassung und Diskussion	30

1 Einführung

Beim räumlichen Schließen, als Teilbereich der Kognitionswissenschaft, wird untersucht, wie räumliche Informationen und Beschreibungen von abstrakten oder realen Objektanordnungen beim Menschen kognitiv verarbeitet werden. Im Alltag ist der Mensch sehr oft mit relationalen Beschreibungen von räumlichen und temporalen Gegebenheiten konfrontiert, ohne großartig Notiz davon zu nehmen. Bei Aufbauanleitungen für Möbelstücke etwa wären wir ohne ein räumliches Verständnis für die beschriebenen Arbeitsschritte komplett verloren. In jedem Schritt müssen ein oder mehrere Teilstücke sortiert und zu neuen, größeren Stücken zusammengesetzt werden, was, je nach Qualität der Anleitung, sich als mehr oder minder schwierig herausstellt. Das Computerspiel *Tetris* zeigt die Leistungsfähigkeit des Gehirns bei der Anordnung räumlicher Objekte. Bei dem Spiel geht es um die Anordnung herabfallender zwei- oder drei-dimensionaler Spielsteine. Die Vorausplanung ist dabei entscheidend für ein maximales Spielergebnis. Geübte Spieler erreichen schier unglaubliche Spielgeschwindigkeiten. Dies ist nur möglich, wenn die aktuelle Spielsituation sehr schnell analysiert wird und die Positionen für die nächsten Spielsteine gut gewählt werden. Was in diesem Fall *gut* bedeutet, möchte ich hier nicht definieren, die Bewertung hängt jedoch von den Folgezuständen nach der Platzierung ab. Um also eine gute Position zu finden, müssen die räumlichen Gegebenheiten der Folgezustände im voraus erkannt werden.

Ein weiteres Beispiel sind Wegbeschreibungen mittels markanten Wegpunkten wie etwa

Die Straße *entlang* und an der großen Ampelkreuzung *links* abbiegen. Im Kreisverkehr *rechts* abbiegen und *über* die Brücke fahren. Danach *zwischen* zwei großen Statuen hindurch und Sie haben Ihr Ziel erreicht.

Der gleiche Sachverhalt ließe sich auch quantitativ über räumliche Koordinaten und Entfernungen beschreiben. Dazu ist jedoch eine komplette Übersicht der Umgebung und natürlich eine genaue Kenntnis der jeweils aktuellen Position nötig. Durch die qualitative Beschreibung entsteht eine räumliche Karte, die nicht unbedingt und sehr genau den echten räumlichen Gegebenheiten entspricht, jedoch die wichtigen Wegmarken intuitiv miteinander verbindet und das Problem der Wegfindung somit vereinfacht und greifbarer macht.

Die obigen Beispiele zeigen, dass der Mensch in der Lage ist, komplexe räumliche Beschreibungen zu verstehen, zu analysieren und neue Informationen aus ihnen zu gewinnen. Das Verständnis dieser Vorgänge ist nicht nur für Kognitionswissenschaftler und Psychologen interessant, sondern kann auch im Bereich der künstlichen Intelligenz Anwendung finden.

Denkbar sind KI-Systeme, die mit menschlichen Beschreibungen räumlicher Gegebenheiten umgehen können oder Pfadplanungen anhand solcher Beschreibungen vornehmen. Ein besseres Verständnis und eine mögliche Simulation dieser Prozesse könnte auch die Mensch-Maschine-Kommunikation im Bereich geographischer Informationssysteme verbessern.

In verschiedenen experimentellen Untersuchungen wurde der menschliche Raumeduktionsprozess mit unterschiedlichen Versuchsanordnungen analysiert (vgl. [2, 7, 9, 4]).

Bei den Experimenten von Ragni et al. wurde der Deduktionsprozess des Menschen an Hand einfacher Objekt-Lage-Beziehungen untersucht [7]. Dazu sollten Versuchspersonen eine Prämissenmenge wie z.B.

A ist links von B
A ist links von C
D ist rechts von B

mental, d.h. ohne Zuhilfenahme von Schreibwerkzeugen oder realen Objekten, verarbeiten und für eine gegebene Konklusion wie etwa

Ist D immer rechts von C?

entscheiden, ob diese in allen möglichen Modellen der Prämissenmenge gilt. Diese Verifikationsaufgaben sollten zeigen, inwieweit die Versuchspersonen in der Lage waren, neues, bisher unbekanntes Wissen aus den gegebenen Informationen zu extrahieren und zu verifizieren. Für diese Prämissen existieren die drei verschiedenen Modelle

ABCD ABDC ACBD.

Das Modell ABDC erfüllt alle Prämissen, widerspricht jedoch der gegebenen Konklusion, welche somit nicht in allen konsistenten Modellen gilt. Doch wie lösen Menschen solche Probleme?

In den Experimenten ging man der der Frage nach, wie die Prämisseninformationen verarbeitet und mental repräsentiert werden. Die Ergebnisse der Untersuchungen gaben Aufschlüsse über die Konstruktion mentaler Modelle und den Ablauf des Deduktionsprozesses. Wie auch in anderen Untersuchungen zeigte sich, dass verschiedene Versuchspersonen gleiche Modelle, sog. Präferierte Mentale Modelle, aus indeterminierten Prämissenmengen erzeugten.

Darauf aufbauend entwickelten Ragni, Knauff und Nebel ein Berechnungsmodell, das *Spatial Reasoning Model* (SRM), um diese Prozesse möglichst genau simulieren und reproduzieren zu können[8]. Die Funktionsweise dieses Modells entspricht einer zweidimensionalen Turingmaschine, die die Prämissenobjekte in einem unendlichen Grid anordnet. Die so entstandenen Modelle können über lokale Transformationen durch einen Fokus verändert werden. Durch den Turingmaschinen-Ansatz ist es möglich, über ein Einheitskostenmaß Komplexitätsanalysen der entwickelten Algorithmen durchzuführen. Das dort vorgestellte Konstruktionsprinzip kann verschiedene Effekte des relationalen Schließens erklären. Insbesondere begründet es die Konstruktion präferierter Modelle über die Anzahl benötigter Operationen während der Modellkonstruktion, die dort am geringsten sind.

Die Algorithmen, die im SRM-Modell verwendet werden, integrierten bisher nur Objekte über 2-stellige Relationen. Zur kognitiven Verarbeitung von komplexeren höherwertigen Relationen gibt es bisher keine experimentellen Untersuchungen.

Der Logiker Charles Peirce beschäftigte sich unter anderem mit der Komposition und Reduktion von Relationen, um neue komplexere Relationen zu definieren oder sie auf einfachere Basisrelationen zurückzuführen [2]. Dieses Konzept soll hier verwendet werden, um den räumlichen Deduktionsprozess komplexer Relationen auf einfachere Basisrelationen zurückzuführen. Auch in der Arbeit von Scivos und Nebel über das ternäre \mathcal{LR} -Kalkül, werden Basisrelationen zur Schaffung neuer Relationen benutzt [12]. Die Basis dieses Kalküls besteht dort aus neun Relationen.

Untersuchungen zur Negation beim räumlichen Schließen stützen die These Relationszerlegung [10]. Danach werden negierte Relationen in eine Menge nicht-negierter Ausdrücke umgeformt und weiterverarbeitet.

Die Basisrelationen müssen bestimmte Eigenschaften erfüllen, die später in dieser Arbeit erörtert werden. Diese Relationen müssen zum Beispiel atomar sein, d.h. sie dürfen nicht in weniger komplexe Relationen zerlegbar sein. Komplexe Relationen entstehen dann aus Kompositionen von Basisrelationen.

Ein Beispiel für eine komplexe Relation ist die 3-stellige Relation *zwischen*. Die Aussage „Y ist zwischen X und Z“ kann in einem zweidimensionalen Grid durch die in Abb. 1 ge-

zeigten Anordnungen modelliert werden. Hierbei existieren je zwei horizontale und vertikale Anordnungen.

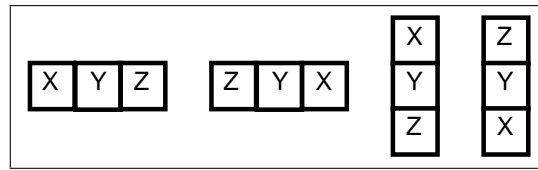


Abbildung 1: 2-dimensionale Anordnungen von Objekten mittels der Relation *zwischen*

Jede dieser Anordnungen kann mit einfachen zweistelligen Relationen beschrieben werden. Die erste etwa lässt sich beschreiben durch die Prämissen

X ist links von Y, Y ist links von Z.

Die Verarbeitung zweistelliger Relationen kann mit dem SRM-Modell gut erklärt werden [8]. Die Frage ist, ob der Mensch höherwertige Relationen auch in Beschreibungen einfacherer Relationen übersetzt oder nicht. Jahn, Johnson-Laird und Knauff haben empirisch untersucht, wie komplexe mehrdeutige Relationen verarbeitet werden [5]. Allerdings wurden die Experimente nur für die zweistellige Relation *neben* und die drei-stellige Relation *zwischen* durchgeführt. Dabei wurden einige Effekte beim Deduktionsprozess (z.B. *order of mention*) festgestellt, der genaue Ablauf der Verarbeitung bleibt jedoch weiterhin unklar. Das Ziel dieser Studienarbeit ist es, ein theoretisches Rahmenwerk einzuführen, wie Deduktionsprozesse beim räumlichen Schließen, durch Zerlegungen in Basisrelationen erklärt werden können. Die formalen Ergebnisse werden in das bestehende SRM-Modell integriert und Algorithmen für die Verarbeitung der Relationen entwickelt. Es soll weiterhin ermöglicht werden, eigene Relationen und deren mögliche Interpretationen zu definieren, die innerhalb des SRM-Modell benutzt und verarbeitet werden können.

Im theoretischen Teil der Arbeit werden kurz die Grundlagen des relationalen räumlichen Schließens und der präferierten mentalen Modelltheorie (PMMT) vorgestellt und das SRM-Modell von Ragni näher erläutert. Danach wird der räumliche Deduktionsprozess und die Konstruktion mentaler Modelle aus formaler Sicht betrachtet und die Grundlage für die Definition und Verarbeitung allgemeiner Relationen geschaffen. Darauf aufbauend wird im dritten Abschnitt die Implementation in das SRM-Modell erläutert. Die Verarbeitungsalgorithmen werden detailliert betrachtet und an Hand von Pseudocode und Beispielen erklärt. Anschließend werden die vorgestellten Ergebnisse im letzten Abschnitt zusammenfassend dargestellt und diskutiert.

2 Theoretische Grundlagen

2.1 Mentale Modelltheorie

Räumliches Schließen beinhaltet meist eine Verarbeitung von relationalen Beschreibungen. Psychologen studieren diesen Prozess schon seit langem (Hunter, 1957), jedoch variieren die verschiedenen Modellierungsansätze. Die beiden wichtigsten Ansätze sind die regelbasierten und die modellbasierten Ansätze. Regelbasierte Ansätze zur relationalen Deduktion versuchen, über bestimmte Ableitungsregeln, die sich aus den logischen Eigenschaften der Relationen ergeben, neues Wissen zu generieren und Schlußfolgerungen logisch abzuleiten [3]. Über die Tiefe der Ableitungsbäume wird auf die kognitive Komplexität, d.h. die Schwierigkeit, die Menschen bei solchen Aufgaben haben, geschlossen. Neben der Komplexität wird versucht, weitere Effekte des qualitativen räumlichen Schließens zu erklären, die in Experimenten auftraten.

Modellbasierten Ansätzen liegt eine grundsätzlich andere Idee zu Grunde. Nach diesem Ansatz wird die in den Relationen enthaltene Information in ein räumliches Modell integriert [7]. Der Inferenzprozess erfolgt nur noch über die im Modell integrierte Information und ist von den ursprünglichen Relationen vollständig entkoppelt. Diese von Johnson-Laird und Byrne (1991) eingeführte *Mentale Modelltheorie* (MMT) ist bis heute der populärste Ansatz, relationales Schließen beim Menschen zu erklären. Goodwin und Johnson-Laird definierten zwei wichtige Eigenschaften mentaler Modelle [2]. Erstens sollte die Struktur dieser Modelle so *ikonisch* wie möglich sein. D.h. die im Modell enthaltenen Entitäten sollten einem mentalen Abbild der Realität oder einer Abstraktion dessen entsprechen. Die zweite Eigenschaft ist das Prinzip der *emergenten Konsequenzen*. Aus diesem folgt, dass Schlußfolgerungen aus relationalen Beschreibungen nur aus den mental modellierten Informationen entstehen.

Versuchsergebnisse zeigen eine inter-individuelle Ähnlichkeit der mentalen Modelle. Verschiedene Personen haben in Experimenten gleiche, und für die einzelnen Relationen typische Modelle konstruiert (siehe [9] und [7]). Diese Modelle werden als Präferierte Mentale Modelle (PMM) bezeichnet.

Des Weiteren konnten verschiedene Effekte bei der räumlich-relationalen Deduktion nachgewiesen werden, die die MMT und die Konstruktion von präferierten mentalen Modellen stützen [9]. So hängt zum Beispiel das resultierende PMM von der Reihenfolge der präsentierten Prämissen ab (Prämissenreihenfolgeeffekt). Weiterhin spielt auch der Aufbau der Prämissen eine wichtige Rolle. So sind etwa negierte Prämissen im Allgemeinen schwieriger zu verarbeiten [10].

Die MMT beschreibt den relationalen Inferenzprozess über drei Phasen. In der ersten Phase, der Modellkonstruktion, werden die gegebenen Prämissen semantisch verarbeitet und deren Information in ein mentales Modell integriert. In der daran anschließenden Modellinspektionsphase, wird das konstruierte Modell auf neues Wissen untersucht, das nicht explizit aus den Prämissen hervorgeht. Über den Ablauf der letzten Phase, der Modellvariationsphase, existieren verschiedene Sichtweisen in der Literatur. In dieser Phase werden alternative Modelle konstruiert, um einer zu verifizierenden Konklusion zu widersprechen. Bisher nahm man an, dass diese drei Phasen separat ablaufen. Alternativ wird der Variationsprozess als Iteration von Modellkonstruktion und -inspektion gesehen. Rauh, Schlieder und Knauff dagegen verwerfen die These der iterativen Konstruktion und Inspektion zugunsten eines Transformationsprozesses, der das konstruierte Modell (PMM) anhand lokaler Transformationen in alternative Modelle überführt [9]. Diese Verfeinerung führte zur *Präferierten Mentalen Modelltheorie* (PMMT).

Die Phasen des Inferenzprozesses sowie die Grundsätze, die in der PMMT definiert werden, konnten empirisch nachgewiesen werden. Der Ablauf der einzelnen Phasen jedoch ist bisher noch nicht vollständig geklärt. Rauh, Schlieder und Knauff haben eine kognitive Modellierung vorgestellt, die es ermöglicht, aus relationalen Intervallbeschreibungen (Allen), präferierte mentale Modelle zu konstruieren [9, 11]. Die Intervalle wurden dabei in Start- und Endpunkte aufgeteilt, die im Konstruktionsprozess, bzgl. der Relationen, in die richtige Reihenfolge zu bringen sind. Dabei wird ein *räumlicher Verarbeitungsfokus* eingeführt, der die Punkte¹ in einer eindimensionalen Raumstruktur anordnet.

Ein weiterer Ansatz zur kognitiven Modellierung dieser Prozesse ist das *Spatial Reasoning Model* (SRM) von Ragni [8]. Dieses Modell versucht einerseits, psychologische Aspekte des räumlichen Schließens basierend auf dem Arbeitsgedächtnismodell von Baddeley (1999) mit einzubeziehen und andererseits komplexitätstheoretische Analysen der Inferenzprozesse zu ermöglichen. Auch hier wird ein Fokus eingeführt, der auf einer zweidimensionalen Gridstruktur arbeitet.

In den bisherigen Arbeiten zur modellbasierten Verarbeitung räumlicher Relationen werden sehr einfache oder sehr spezielle Relationen behandelt. Das o.g. Modellierungskonzept von Rauh et al. verarbeitet nur Intervallrelationen. Auch in den Arbeiten von Johnson-Laird werden meist nur 2-stellige Relationen behandelt, die, separat betrachtet, keine indetermierte Interpretation besitzen. In der Praxis werden jedoch auch diese häufig benutzt. Bei der Aussage *A ist neben B* etwa ist nicht eindeutig festgelegt, wie die beiden Objekte zueinander liegen. *A* könnte links oder rechts neben dem Objekt *B* platziert werden. Ein etwas komplexeres Beispiel ist die 3-stellige Relation *zwischen*. Die Aussage *A ist zwischen B und C* etwa beschreibt nicht nur eine indetermierte Anordnung, sondern bezieht sich gleich auf 3 Objekte. Zu solch komplexen Relationen gibt es bisher keine modellbasierten Ansätze zur Simulation der o.g. Inferenzphasen.

Der Mathematiker und Philosoph Charles Peirce hat unter anderem die Komposition und Reduktion von Relationen untersucht. Diese Erkenntnisse können dazu verwendet werden, komplexe Relationen wie *zwischen* und *neben* in logische Ausdrücke einfacher Relationen zu überführen. Als einfache Relationen werden in dieser Arbeit die Relationen *links*, *rechts*, *vor*, und *hinter* verwendet. Die Aussage *A ist neben B* kann dann mittels der xor-Verknüpfung der Aussagenlogik zu folgendem Ausdruck überführt werden:

$$A_{links} B \oplus A_{rechts} B \oplus A_{hinter} B \oplus A_{vor} B$$

Das Ziel dieser Arbeit ist es, räumliche Inferenzprozesse komplexer Relationen mit dem SRM zu realisieren. Dazu werden zunächst komplexe Relationen systematisiert und in die bisherige Modellierung der Konstruktions-, Inpektions- und Variationsphase integriert. Die bisher entwickelten Algorithmen werden dahingehend erweitert. Eine flexible Beschreibungsmöglichkeit von Relationen soll es außerdem ermöglichen, komplexe Relationen selbst definieren zu können, um so eine experimentelle Basis für weitere Untersuchungen bezüglich des räumlichen Schließens zu bieten. Die kognitive Simulation soll zwar möglichst äquivalent zum menschlichen Schließen sein, jedoch wurden dazu keine Experimente unternommen. Die hier vorgestellten Verarbeitungsalgorithmen basieren auf der bekannten Verarbeitung einfacher Relationen.

¹Die Punkte sind als räumliche Entitäten zu betrachten.

2.2 SRM

Das von Ragni entwickelte SRM [8] gliedert sich in vier Subsysteme (Abb. 2). Zentraler Bestandteil ist ein *Kontrollprozess*, der die Verarbeitung und Kommunikation mit den anderen Subsystemen steuert. Der *Semantische Interpretierer* verarbeitet Prämisseneingaben in Form von natürlicher Sprache und übersetzt diese in eine interne Repräsentation. Diese Informationen werden in den räumlich-visuellen Speicher integriert. Dieser Speicher entspricht einer zweidimensionalen Zellenstruktur (Spatial Array). Auf dieser Struktur arbeitet ein Fokus, der, ähnlich dem Schreib-/Lesekopf einer Turingmaschine, eine kleine Menge an Funktionen unterstützt. Dazu gehören Bewegungsoperationen und Operationen, die Objektentitäten in die Zellen schreiben und aus ihnen lesen kann.

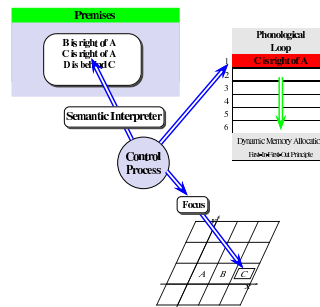


Abbildung 2: Spatial Reasoning Model. Die Information der Prämissen wird in den räumlich-visuellen Buffer mittels eines Fokus abgelegt. Im hier indeterminierten Fall der Prämisse „C rechts A“ wird die zusätzliche Information als Annotation in der artikulatorischen Schleife gehalten.

Um in Fällen mit indeterminierter Information alternative aber dennoch konsistente Modelle erzeugen zu können, dient die *Artikulatorische Schleife*. Sie entspricht einer optional begrenzten Speicherstruktur, in der indeterminierte Prämissen als Annotationen gehalten werden. Im Fall eines begrenzten Speichers, können verschiedene Verdrängungsstrategien (Least Recently Used, Least Frequently Used, usw.) angewendet werden, um Informationen im Falle eines Überlaufs zu entfernen.

Für die Konstruktion der PMMs werden sukzessive alle Prämissen eines Deduktionsproblems verarbeitet und die Information der Objektentitäten in das Spatial Array integriert. Die Konstruktionsphase unterscheidet dazu folgende Prämissentypen der Form $x R y^2$:

1. Initiale Prämisse: Die erste Prämisse des Deduktionsproblems.
2. 1-Objekt-Prämissen: Prämissen bei denen entweder x oder y ins Modell eingefügt werden muss.
3. 2-Objekt-Prämissen: Prämissen bei denen sowohl x als auch y noch nicht im Modell enthalten sind. Da die Objekte dieser Prämisse nicht zu den bisher im Modell enthaltenen Objekte in Relation gesetzt werden können, werden sie in ein neues Spatial Array eingefügt.

² y wird als Referenzobjekt (RO) bezeichnet, x als „zu lokalisierendes Objekt“ (LO)

4. Verknüpfungsprämissen: Prämissen bei denen x und y in verschiedenen Spatial Arrays enthalten sind. Beide Arrays werden über solche Prämissen verschmolzen.
5. 0-Objekt-Prämissen: Prämissen, deren Argumente bereits im Modell enthalten sind. Die Verarbeitung dieser Prämissen wird auch als *belief revision* bezeichnet.

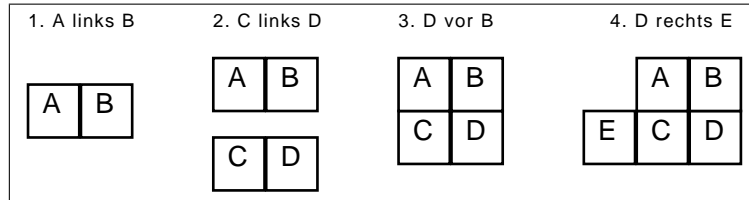


Abbildung 3: Modellkonstruktion für die Prämissen A links B , C links D , D vor B und D rechts E .

Abbildung 3 zeigt den Konstruktionsprozess für eine Prämissenmenge, die bis auf 0-Objekt-Prämissen alle o.g. Prämissentypen enthält. Bei der Verarbeitung der initialen Prämisse wird zuerst das Referenzobjekt B ins Modell eingefügt. Danach bewegt sich der Fokus in Richtung der angegebenen Relation *links* und fügt an die freie Position das Objekt A ein. Die zweite Prämisse ist eine 2-Objekt-Prämisse, die beide Objekte in ein neues Teilmodell integriert. Durch die dritte Prämisse werden diese beiden Teilmodelle zu einem Modell verbunden. Dabei werden die Teilmodelle bzgl. der Relation *vor* an den Objekten D und B ausgerichtet. Die letzte Prämisse ist eine 1-Objekt-Prämisse. Hierbei muss beachtet werden, ob das LO oder RO eingefügt werden muss. Dies entscheidet über die Fokusbewegungsrichtung beim Suchen einer freien Position. Hier wird der Fokus auf das schon enthaltene Objekt D bewegt. Da D das LO ist, muss der Fokus in die entgegengesetzte Richtung, hier also nach links, bewegt werden, um eine freie Position für E zu suchen.

Diese Einfügestrategie wird als *first-free-fit*-Prinzip bezeichnet. Des Weiteren wird noch das *first-fit*-Prinzip unterschieden, bei dem das Objekt in die nächstmögliche Position eingefügt wird und evtl. schon vorhandene Objekte verschoben werden. Prämissen, deren Objekte nicht direkt nebeneinander platziert werden können, werden als Annotation in der artikulatorischen Schleife gehalten. Der Variationsprozess greift auf diese Informationen zurück, um aus dem konstruierten Modell alternative Modelle variieren zu können.

In der zweiten Phase, der Inspektionsphase, wird das konstruierte Modell bzgl. einer gegebenen Prämisse inspiziert. Dazu wird der Fokus benutzt, um die Positionen der Prämissenobjekte zu finden und deren Lage zueinander zu berechnen.

Der Variationsprozess als dritte und letzte Deduktionsphase versucht, eine gegebene Konklusion zu verifizieren und ggf. Gegenbeispiele zu erzeugen, die der Konklusion widersprechen, jedoch konsistent mit der ursprünglichen Modellbeschreibung sind. Dies geschieht über lokale Transformationen von annotierten Objekten, da deren Platzierung indeterminiert ist. Die LOs der Annotationen definieren die annotierten Objekte. Die ROs werden als *Anker* bezeichnet, über die annotierte Objekte nicht hinausbewegt werden dürfen. Über rekursive Beziehungen von Ankerobjekten ist eine rekursive Variation möglich. Dies ermöglicht die Variation von Objekten, die nicht direkt miteinander in Beziehung gesetzt werden können. Die bisher im SRM zur Verfügung stehenden Relationen beschränken sich auf die Relationen *Links*, *Rechts*, *Vor*, und *Hinter*. In den folgenden Abschnitten sollen theoretische

Überlegungen erörtert werden, den Raum der Relationen zu erweitern und eine Verarbeitung komplexer Relationen mit dem SRM zu ermöglichen. Da auch dieser Ansatz Einschränkungen bzgl. der Relationen vornimmt, werden die Grenzen der Beschreibungsmöglichkeiten im darauf folgenden Abschnitt diskutiert.

2.3 Formale Aspekte des relationalen Schließens

Das SRM verarbeitet sequentiell eine geordnete Menge relationaler Ausdrücke. Diese Ausdrücke stellen eine Teilmenge von S-Ausdrücken (nach [1]) dar.

Definition 1. *Ein SRM-Ausdruck ist ein S-Ausdruck, der durch folgende Regeln gebildet wird:*

1. Sind t_1, \dots, t_n S-Terme und ist R ein n -stelliges Relationssymbol einer Symbolmenge S , so ist $Rt_1 \dots t_n$ ein SRM-Ausdruck.
2. Sind φ und ψ SRM-Ausdrücke, so auch $(\varphi \wedge \psi)$.

Analog zur Definition in [1] bezeichnet L^{SRM} die Menge aller möglichen Ausdrücke, also die Sprache des SRM. Die Struktur dieser Ausdrücke beschränkt sich nach dieser Definition auf Konjunktionen von relationalen Ausdrücken.

Die Semantik von SRM-Ausdrücken ergibt sich erst bei Interpretation über einer bestimmten S-Struktur. Da die Modellierung in einer zweidimensionalen Gridstruktur stattfinden soll, wird \mathbb{Z}^2 als Grundbereich der Struktur gewählt. Dort wird jeder Objektentität eine eindeutige Position in Koordinatenform zugewiesen.

Des Weiteren wird die Funktion d für die Berechnung der Manhattan-Distanz zwischen zwei Positionen eingeführt. Diese dient dazu, Distanzen zwischen Objekten bei der Relationsdefinition festzulegen (siehe Abschnitt 2.5ff). Die SRM-Struktur ergibt sich dann zu

$$\mathfrak{S} = (\mathbb{Z}^2, L, R, V, H, d).$$

L, R, V und H stehen für die zweistelligen Relationen *links*, *rechts*, *vor* und *hint*er die sich folgendermaßen definieren:

$$\begin{aligned} \text{(links)} \quad L &= \{(a, b) = ((a_x, a_y), (b_x, b_y)) \mid a_x < b_x \text{ und } a_y = b_y\} \\ \text{(rechts)} \quad R &= \{(a, b) = ((a_x, a_y), (b_x, b_y)) \mid a_x > b_x \text{ und } a_y = b_y\} \\ \text{(vor)} \quad V &= \{(a, b) = ((a_x, a_y), (b_x, b_y)) \mid a_x = b_x \text{ und } a_y < b_y\} \\ \text{(hintere)} \quad H &= \{(a, b) = ((a_x, a_y), (b_x, b_y)) \mid a_x = b_x \text{ und } a_y > b_y\} \end{aligned}$$

Das SRM versucht, ein Modell für eine geordnete Prämissenmenge $\Phi = \{\varphi_1, \dots, \varphi_n\} \subset L^{SRM}$, $n \in \mathbb{N}$ zu konstruieren. Auf Φ ist eine Ordnung definiert, um die Prämissenreihenfolge beizubehalten. Aus Sicht der mathematischen Logik entspricht dieser Prozess dem Finden einer Belegung β , so dass die Interpretation $\mathfrak{I} = (\mathfrak{S}, \beta)$ ein Modell von Φ ist, also $\mathfrak{I} \models \varphi$ für alle $\varphi \in \Phi$.

Bei 1-Objekt-Prämissen, bei denen nur das LO einzufügen ist, muss der Fokus in die entgegengesetzte Richtung der Relation nach einer freien Position für das Objekt suchen. Dies ist dann der Fall, wenn das LO schon im Modell enthalten ist und das Referenzobjekt der Prämisse eingefügt werden muss. Intuitiv ist klar, dass wenn A *links* B gilt, so auch B *rechts* A gelten muss. Diese Umkehrung einer Relation wird als *konverse Relation* bezeichnet. Für eine Relation R wird sie mit R^{-1} angegeben.

Aus den Definitionen der o.g. Relationen gehen die Beziehungen zu den konversen Relationen unmittelbar hervor: Es gilt:

$$\text{Links}(a, b) \models \text{Links}^{-1}(b, a) \models \text{Rechts}(b, a) \models \text{Rechts}^{-1}(a, b) \quad (1)$$

$$\text{Hinte}(a, b) \models \text{Hinte}^{-1}(b, a) \models \text{Vor}(b, a) \models \text{Vor}^{-1}(a, b). \quad (2)$$

Das Problem der Modellkonstruktion kann auch als Constraint Satisfaction Problem (CSP) aufgefasst werden. Eine Prämissenmenge $\{\varphi_1, \dots, \varphi_n\}$, $n \in \mathbb{N}$, mit $\varphi_i(a_i, b_i) = Ra_i b_i$ stellt dabei eine Menge von Constraints für das zu konstruierende Modell dar. Die Menge der CSP-Variablen ergibt sich aus den in den Prämissenausdrücken enthaltenen Variablen zu:

$$V = \bigcup_{i=1}^n \{a_i, b_i\} \subset \mathbb{Z}^2.$$

Die unendliche Domäne \mathbb{Z}^2 der CSP-Variablen kann auf eine endliche Domäne beschränkt werden, da die Anzahl der Prämissen und Objekte beschränkt ist, und somit keine unendlich großen Modelle entstehen können.

O.B.d.A. kann angenommen werden, dass die Modellkonstruktion im Koordinatenursprung $(0,0)$ beginnt, d.h. dort das erste Objekt platziert wird. Wenn nun mit jeder weiteren Prämisse ein weiteres Objekt in das Modell integriert wird, müssen bei $n \in \mathbb{N}$ Prämissen insgesamt n Objekte eingefügt werden. Daraus ergibt sich eine maximale Modellausdehnung von $n + 1$ Feldern, wenn das Modell linear aufgebaut ist. Weil sich das Modell am Koordinatenursprung beginnend aufgebaut wird ergibt sich für die CSP-Variablen die endliche Domäne

$$D = \{(x, y) \in \mathbb{Z}^2 \mid -n \leq x, y \leq n \text{ mit } n = |V|\}.$$

Um aus einer Prämissenmenge nun ein Modell zu erzeugen, werden sukzessive die Objekte aller relationalen Ausdrücke in die Zellenstruktur eingefügt. D.h. den Objekten werden Koordinaten zugewiesen, die einer erfüllenden Belegung entsprechen. Dies geschieht nicht direkt, sondern über den Verarbeitungsfokus, der nach freien Objektpositionen im Grid sucht. Das erste zu verarbeitende Objekt, ist das RO der initialen Prämisse. Es wird mit der Koordinate $(0,0)$ belegt. Die Belegungen für die folgenden Objekte werden durch bewegen des Fokus in Richtung der Prämissenrelationen gesucht. Die Bewegung des Fokus entspricht einer Änderung der aktuellen Fokus-Koordinaten. Die Positionsänderung erfolgt dabei in der horizontalen oder vertikalen Ebene (siehe Abb. 4).

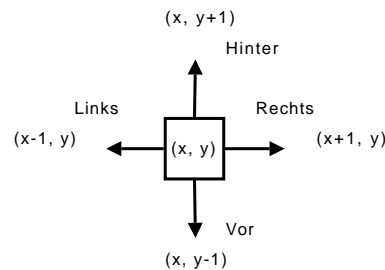


Abbildung 4: Fokuspositionsänderungen für alle möglichen Fokusbewegungsrichtungen

Bei 0-Objekt-Prämissen ist es evtl. nötig, das bisher konstruierte Modell zu variieren. Den platzierten Objekten werden dann rekursiv neue Positionen durch Vertauschungen zugewiesen (siehe [8]). Dieser Prozess ist vergleichbar mit dem Backtracking in CSPs.

2.4 Einfache und komplexe Relationen

Die o.g. Relationen, die bisher im SRM verarbeitet werden können, werden im Folgenden als Basisrelationen verwendet, um daraus komplexere Relationen zu komponieren. Diese Relationen entsprechen den möglichen Fokusbewegungsrichtungen und zeichnen sich durch eine einfache Verarbeitung aus. Die Definitionen der horizontalen Relationen *Links* und *Rechts*, sowie der vertikalen Relationen *Hinter* und *Vor* können logisch äquivalent ineinander überführt werden.

Die festgelegten Basisrelationen besitzen alle die Eigenschaft der Transitivität. Sie sind nicht symmetrisch und nicht reflexiv. Die Forderung einer konversen Relation ist nötig, um Prämissen, deren Referenzobjekt eingefügt werden muß, verarbeiten zu können.

Mit n -stelligen komplexen Relationen ($n \geq 2$) soll es ermöglicht werden, kompliziertere Sachverhalte auszudrücken, für die mehrere Basisausdrücke nötig wären oder die, wie im Fall der *neben*-Relation, mehrdeutig sind.

Die verschiedenen Interpretationsmöglichkeiten einer Relation werden dafür in Form von SRM-Ausdrücken aufgezählt, die über die Exklusiv-Oder-Verknüpfung zusammengesetzt sind. Eine Relation gilt also genau dann, wenn einer der SRM-Ausdrücke gilt.

Definition 2. Für eine endliche Menge $\{\varphi_1, \dots, \varphi_m\} \subset L^{SRM}$ von SRM-Ausdrücken, in denen nur die Variablen $a_1, \dots, a_n, n \in \mathbb{N}, n \geq 2$, vorkommen, definiert sich die komplexe n -stellige Relation R_c zu:

$$R_c = \{(a_1, \dots, a_n) \mid \bigoplus_{i=1}^m \varphi_i\}.$$

Eine komplexe Relation ist also erfüllt, wenn genau einer der SRM-Ausdrücke erfüllt ist.

Im folgenden werden die SRM-Ausdrücke, die eine komplexe Relation definieren als Interpretationen bezeichnet, die nicht mit Interpretationen der mathematischen Logik zu verwechseln sind.

Durch die Einführung komplexer Relationen wird es ermöglicht, neue, selbst definierte Relationen und vor allem komplexere Relationen im SRM zu verarbeiten. Die Definition über SRM-Ausdrücke erlaubt es, durch eine rekursive Zerlegung, die Verarbeitung dieser Ausdrücke auf einfache Basisrelationen zurückzuführen.

Komplexe Relationen können, wie im Beispiel der Relation *zwischen* eine Verarbeitung von mehr als zwei Objekten und Richtungswechsel des Fokus erfordern. Da die Verarbeitung der Relationen auf die Verarbeitung zweistelliger Relationen zurückgeführt wird, müssen die verwendeten SRM-Ausdrücke einer weiteren Bedingung genügen.

Für die SRM-Ausdrücke $\varphi_1, \dots, \varphi_m$, einer n -stelligen komplexen Relation $R_c(a_1, \dots, a_n)$, juss gelten:

$$\varphi_i = \bigwedge_{\substack{j,k \in \{a_1, \dots, a_n\} \\ j \neq k}} R a_j a_k \quad \text{mit } R \in \{L, R, F, B\}.$$

D.h., für eine Relationsdefinition ist es erforderlich, sämtliche zweistelligen Kombinationen der Relationsargumente für die Angabe Lageinterpretation anzugeben. Dies ist nötig, weil vor der Verarbeitung der Relation im SRM nicht bekannt ist, welche Objekte schon im Modell enthalten sind.

Beispiel 1. Die zwischen-Relation R_{zw} wird mit allen vier möglichen Interpretationen (siehe Abb. 1) wie folgt definiert:

$$R_{zw} = \{(a, b, c) \mid \varphi_1 \oplus \varphi_2 \oplus \varphi_3 \oplus \varphi_4\} \quad \text{mit}$$

$$\begin{aligned} \varphi_1 &= a \text{ links } b \wedge b \text{ links } c \wedge a \text{ links } c \\ \varphi_2 &= c \text{ links } b \wedge b \text{ links } a \wedge c \text{ links } a \\ \varphi_3 &= a \text{ hinter } b \wedge b \text{ hinter } c \wedge a \text{ hinter } c \\ \varphi_4 &= c \text{ hinter } b \wedge b \text{ hinter } a \wedge c \text{ hinter } a \end{aligned}$$

Durch diesen Ansatz können nur Relationen beschrieben werden, deren endgültige Objektanordnung logisch äquivalent durch eine aus Basisrelationen bestehende Prämissenmenge beschreibbar ist. Dies mag auf den ersten Blick sehr restriktiv erscheinen, jedoch wird der Raum der Relationen, die mit dem SRM verarbeitet werden können, erheblich erweitert.

2.5 Komplexe Relationen in der SRM-Software

Um die formal definierten Relationen in der Implementation des SRM nutzen zu können, werden sie in XML-Dateien beschrieben. Diese sind relativ leicht zu beschreiben und können leicht ausgewertet und auf korrekte Syntax geprüft werden. Des Weiteren ist dieses Format leicht erweiterbar. Sämtliche definierbaren Parameter sind in Tabelle 1 aufgelistet. Der wichtigste Parameter ist die konjunktive Beschreibung der möglichen Objektanordnungen, da diese für die Interpretation der Relation entscheidend ist. Sowohl horizontale als auch vertikale Anordnungen der Objekte müssen vollständig aufgezählt werden.

Tabelle 1: Benötigte Eigenschaften räumlicher und temporaler Relationen

Eigenschaft	Wertebereich	Beschreibung
name	Zeichenkette	Name der Relation
abbreviation	Zeichenkette	Abkürzender Bezeichner
base_relation	{True, False}	Eigenschaft der Basisrelation
arity	\mathbb{N}	Stelligkeit der Relation
domain	{spatial, temporal}	Domäne der Relation
converse_relation	{ R , None }	Name der konversen Relation, oder False, falls keine existiert.
range	(\mathbb{N}, \mathbb{N})	Verarbeitungsdistanz
conjunctions	$\{\varphi_1, \dots, \varphi_n \mid \varphi_i \in \mathcal{L}(\text{SRM})\}$	Relationsinterpretationen

Als besonderes Attribut wird hier noch das Attribut *range* eingeführt, um eine gewisse Verarbeitungsdistanz für die Relation anzugeben. Dieses Attribut bietet die Möglichkeit, die minimale und maximale Verarbeitungsdistanz des Fokus einzuschränken. Damit ist es möglich, Relationen wie etwa *direkt links* oder *nahe bei* zu definieren. Für die Relation *direkt links* etwa soll eine maximale und minimale Distanz von 1 festgelegt sein.

Für die Basisrelationen sind nur minimale Distanzen von 1 festgelegt. Eine Beschränkung der maximalen Distanz gibt es bei diesen Relationen nicht. Die Basisrelationen sind bisher auch die einzigen, die eine konverse Relation festlegen.

Die einzelnen Relationseigenschaften sollen hier am Beispiel der Relation *links* verdeutlicht werden, deren XML-Beschreibung in Abb. 5 angegeben ist. Der Name der Relation ist *links*, der mit *L* abgekürzt wird. In Prämissenbeschreibungen ist dann sowohl der Name als auch die Abkürzung erlaubt. Bei der internen Verarbeitung wird nur die Abkürzung verwendet. Die Relation ist als 2-stellige, räumliche Basisrelation definiert, deren konverse Relation die Relation *rechts* ist (Zeile 4-7). Das *range*-Tag beinhaltet die minimale und maximale Verarbeitungsdistanz des Fokus, die hier nur nach unten beschränkt ist. Die maximale Distanz von *i* wird intern als unendlich interpretiert.

Zuletzt wird die Semantik der Relation festgelegt. Die Relation wird durch einen einzigen SRM-Ausdruck beschrieben. Der einzige notwendige Term spezifiziert, mit welcher Relation die Argumente verknüpft werden. In diesem Beispiel ist das erste Argument links des zweiten zu platzieren. Bei der Spezifikation der Semantik werden, statt Variablen, ganzzahlige Indizes für die Relationsargumente benutzt.

```

1 | <?xml version="1.0">
2 | <relation >
3 |   <name>links </name>
4 |   <abbreviation>L</abbreviation>
5 |   <base>true</base>
6 |   <arity>2</arity>
7 |   <domain>spatial</domain>
8 |   <converse>rechts </converse>
9 |   <range>
10 |     <min>1</min>
11 |     <max>i</max>
12 |   </range>
13 |   <semantic>
14 |     <conj>
15 |       <term>
16 |         <arg>1</arg>
17 |         <arg>2</arg>
18 |         <rel>left </rel>
19 |       </term>
20 |     </conj>
21 |   </semantic>
22 | </relation >

```

Abbildung 5: XML-Beschreibung der Basisrelation *links*.

Bei den Basisrelationen, die den Fokusbewegungsrichtungen entsprechen, ist diese vermeintlich rekursive Definition notwendig, um intern die neu hinzugefügten Relationsattribute auch für die bisherigen Basisrelationen verwenden zu können. Die Verarbeitung dieser Relationen geschieht wie in der bisherigen Version des SRM (vgl. [8]). Die Definition einer komplexen Relation erfordert meist die Verwendung mehrerer Konjunktionen mit jeweils mehreren Termen. Diese werden nacheinander in der *semantic*-Umgebung aufgezählt. Die Reihenfolge der spezifizierten Konjunktionen legt deren Präferenz für die Verarbeitung fest. Die zuerst bzw. zuletzt spezifizierten Konjunktionen erhalten eine hohe bzw. niedrige Präferenz. Die versuchsweise definierten Relationen *neben* und *zwischen*, die zwangsläufig indeterminiert sind, orientierten sich an dem in [5] empirisch nachgewiesenen Prinzip der *order of mention*.

2.6 Konstruktionsprozess

Die Verarbeitung komplexer Relationen erfordert für den Konstruktionsprozess eine Anpassung der Verarbeitungsalgorithmen. Die Verarbeitung von Basisrelationen bleibt unverändert. Komplexe Prämissen jedoch werden bei der Verarbeitung in Basisrelationen zerlegt und diese auf konventionelle Art in das Modell integriert. Jede komplexe Relation besteht aus mehreren Mengen von SRM-Ausdrücken, die logisch über eine \oplus -Relation verbunden sind. Dadurch wird sichergestellt, dass genau eine dieser Konjunktionen in das Modell integriert wird. Die auf den Konjunktionen definierte Ordnung legt zugleich die Interpretationspräferenz für die Konstruktion der PMMs fest. Beispielsweise kann die Prämisse *A neben B* durch *A links B*, oder (exklusiv) *B links A* interpretiert werden. Dabei soll *A links B* als präferierte Interpretation festgelegt sein³.

Die möglichen Interpretationen komplexer Relationen werden bei der Konstruktion in einem Modellbaum zusammengefasst. Jede Konjunktion wird dabei als ein neuer Ast in diesen Baum eingefügt. Dabei werden die Terme jeder Konjunktion in einer Modellkopie des Elternknotens integriert. Unmögliche oder widersprüchliche Modelle werden während der Konstruktionsphase aus dem Modellbaum entfernt. Daraus ergibt sich ein Baum, dessen Pfade alle dieselbe Länge besitzen. Der Wurzelknoten dieses Baumes ist ein leerer Knoten, dessen Tochterknoten die möglichen Modelle der initialen Prämisse enthalten. Basisrelationen, die nicht zu einer komplexen Relation gehören, werden in den Modellen aller Blätter des aktuellen Modellbaums integriert. Besteht der Baum zum Zeitpunkt der Verarbeitung nur aus dem Wurzelknoten, wird ein leeres Modell erzeugt und die Prämisse dort eingefügt. Da die Objektpositionen bei komplexen Relationen meist indeterminiert sind, werden diese ebenfalls als Annotationen in der artikulatorischen Schleife abgelegt. Die Annotation wird dabei auf die im Modell verwendete Interpretation der Relation reduziert. Der Grund dafür liegt in der beschränkten Verarbeitungskapazität des Menschen, die hier simuliert werden soll. Es wird davon ausgegangen, dass nach Verarbeitung einer Relation mit mehreren Interpretationsmöglichkeiten, die Interpretation nicht mehr verändert wird [5]. Innerhalb des Modellbaums finden also keine Sprünge zwischen den Blättern statt, so dass die Variationen nur im konstruierten PMM stattfinden.

Die Pfade des Baumes spiegeln die Interpretationspräferenz der komplexen Relationen wider. Das Blatt des linken Pfades enthält das PMM für die Ausgangsprämissen. Am weitesten rechts befindet sich das von der Präferenz her unwahrscheinlichste Modell.

Es muss jedoch berücksichtigt werden, dass die Ordnung der Modelle im Baum nicht notwendigerweise mit Ergebnissen aus Experimenten mit Versuchspersonen übereinstimmen muss. Abb. 6 zeigt den Modellbaum für die Prämissen *A neben B*, *B zwischen C und D*.

Die *neben*-Relation besitzt wie o.g. zwei, die *zwischen*-Relation vier mögliche Interpretationen. In diesem Fall existieren nach Verarbeitung der initialen Prämisse zwei mögliche Modelle. Jedes dieser Modelle wiederum besitzt vier Tochtermodelle für die vier Interpretationen der *zwischen*-Prämisse. Das präferierte Modell des rechten Teilbaums (CBAD) unterscheidet sich vom PMM (CABD) nur in der anderen Interpretation der initialen Prämisse. Die zweite Prämisse wurde auf die gleiche Weise interpretiert. Die Modelle unterscheiden sich nur durch die vertauschten Positionen der Objekte *A* und *B*.

Um untersuchen zu können, welche Modelle für Menschen leichter zu konstruieren sind, und welche alternativen Modelle am schnellsten gefunden werden, sind auch andere Metriken für den Modellvergleich denkbar. Da die Objekte im \mathbb{Z}^2 angeordnet werden, könnte zum Beispiel die Manhattan-Distanz aufschlussreicher sein.

³Prinzip der *order of mention*[5]

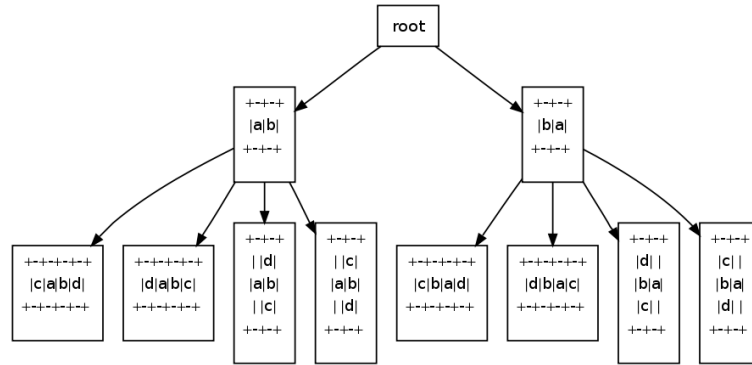


Abbildung 6: Ausgabe des Modellbaums nach Verarbeitung der zwei komplexen Prämissen A neben B und B zwischen C und D . Das linkeste Blatt des Baums stellt das PMM dar.

Definition 3. Für die benutzten Modellstrukturen berechnet sich die Manhattan-Distanz zwischen zwei Modellen m_1 und m_2 , die dieselbe Objektmenge O besitzen, durch:

$$d(m_1, m_2) = \sum_{o \in O} |pos_x(o, m_1) - pos_x(o, m_2)| + |pos_y(o, m_1) - pos_y(o, m_2)|$$

Die Funktionen $pos_x(o, m)$ bzw. $pos_y(o, m)$ liefern jeweils die X - bzw. die Y -Koordinate eines Objekts o bzgl. des Modells m .

Für die in Abb. 6 konstruierten Modelle ergeben sich die in Tabelle 2 berechneten Manhattan-Distanzen. Bei den Modellen des linken Teilbaums befindet sich das Objekt A auf Position $(0, 0)$. Im rechten Teilbaum befindet sich an dieser Position das Objekt B . Es zeigt sich, dass die komplette Spiegelung des PMM (DBAC), die höchste Distanz aufweist. Dagegen besitzt das Modell CBAD, das präferierte Modell des rechten Teilbaums, die geringste Distanz zum PMM.

Tabelle 2: Manhattan-Distanzen aller Modelle aus Abb. 6 bzgl. des PMMs $CABD$.

Modell:	$CABD$	$DABC$	D AB	C AB	$CBAD$	$DBAC$	D BA	C BA
Distanz:	0	6	C 5	D 5	2	8	C 7	D 7

Des Weiteren könnte auch der konzeptuelle und generelle Nachbarschaftsgraph Aufschluß über das Deduktionsverhalten des Menschen bei komplexen Relationen geben, so dass die kognitive Simulation weiter verfeinert werden kann (vgl. [7]).

Es kann angenommen werden, dass der Mensch nicht, oder nur bedingt in der Lage ist, einen kompletten Modellbaum zu erzeugen. Da die Prämissen und Terme jedoch auf Basis experimentell untersuchter Verfahren eingefügt werden, kann jeder Ableitungspfad als kognitive Simulation betrachtet werden. Aufgrund der festgelegten Präferenzen innerhalb der Konjunktionen bleibt somit die kognitiv adäquate Konstruktion⁴ eines PMM erhalten.

⁴i.S. der bisherigen Untersuchungen zum SRM.

Die Betrachtung des kompletten Modellbaums erlaubt genauere Analysen des räumlichen Schließens. Die verschiedenen Ableitungspfade und die komplette Übersicht über die Modelle aller Relationsinterpretationen kann für die Auswertung experimenteller Untersuchungen von großem Nutzen sein.

2.7 Inspektionsphase

Bei der Inspektion eines Modells wird geprüft, ob das konstruierte Modell (\mathfrak{S}, β) eine bestimmte Relation erfüllt, d.h. ob $(\mathfrak{S}, \beta) \models \varphi = R(a, b)$ gilt. Dies geschieht ausschließlich über Fokusoperationen, die entsprechend der Relation gesteuert werden.

Der Fokus wird zuerst auf das Objekt der Relationsargumente bewegt, dass am dichtesten an der aktuellen Fokusposition liegt. Die Bewegung erfolgt dabei ausschließlich innerhalb des Modells, also über belegte Zellen der zu Grunde liegenden Gridstruktur. Da auch nicht-lineare oder ringförmige Modelle möglich sind, wird dazu ein kürzester Pfad durch das Modell mittels des Dijkstra-Algorithmus berechnet (siehe [6]).

Vor der Fokusbewegung wird dafür ein Graph $G = (V, E)$ des Modells erstellt. Die Knotenmenge V besteht aus den im Modell enthaltenen Objekten. Die Kantenmenge ergibt sich aus den adjazent liegenden Objekten:

$$E = \{o_1, o_2 \mid o_1, o_2 \in O, \text{dist}(o_1, o_2) = 1\}$$

Dabei entspricht die Distanzfunktion $\text{dist}(o_1, o_2)$ der Manhattan-Distanz zwischen den Objekten o_1 und o_2 . Die Operationen für die Berechnung des Modellgraphen gehen dabei nicht in die Kostenanalyse der Inspektion ein. Für die Kostenanalyse werden ausschließlich die Fokusoperationen gezählt.

Dieses Verfahren ist in Abb. 7 graphisch dargestellt. Dort soll der Fokus vom Objekt A zum Objekt E bewegt werden. Die Bewegung wäre innerhalb der Gridstruktur in zwei Schritten möglich (gestrichelte Linie). Da der Fokus sich jedoch über die Objekte des Modells bewegen soll, wird die Bewegung über den Modellgraph berechnet. Dadurch ergeben sich insgesamt vier Bewegungsoperationen.

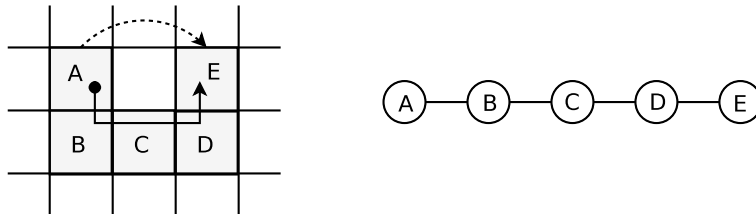


Abbildung 7: Fokusbewegung von Objekt A zum Objekt E über den kürzesten Pfad innerhalb des Modells. Links ist die Objektordnung im Grid dargestellt. Rechts der dazugehörige Modellgraph.

Für die Inspektion wird der Fokus vom aktuell fokussierten Objekt in Richtung der gegebenen Relation so lange bewegt, bis das gesuchte Objekt gefunden ist, oder der Rand des Modells erreicht ist. Wenn die Inspektion beim LO der Relation beginnt, wird der Fokus entsprechend der konversen Relation bewegt. Wird das Objekt gefunden, d.h. der Fokus erreicht eine Zelle mit dem gesuchten Objekt, ist die Inspektion dieser Relation erfolgreich beendet.

Bei komplexen Relationen werden die einzelnen Interpretationen der Präferenz nach abgearbeitet. Jede Interpretation wird in ihre Basisrelationen übersetzt, die nacheinander geprüft werden. Wenn alle Basisrelationen erfolgreich inspiziert wurden, ist auch die Inspektion der komplexen Relation erfolgreich. Da nur eine der Interpretationen gelten kann, wird der Inspektionsprozess abgebrochen, sobald eine Interpretation erfolgreich geprüft wurde.

2.8 Variationsprozess

Bei der Modellvariation, dem eigentlichen Deduktionsprozess, wird das konstruierte PMM bezüglich einer gegebenen Konklusion variiert und inspiziert, um die Gültigkeit einer gegebenen Konklusion in allen Modellen zu prüfen. Dazu werden mittels lokaler Transformationen, alternative Modelle des PMMs erzeugt. Die lokalen Transformationen bestehen aus Vertauschungsoperationen benachbarter Objekte, deren Position indeterminiert ist.

Indeterminierte Objektpositionen entstehen, wenn während der Modellkonstruktion die Position des einzufügenden Objektes belegt ist, oder komplexe Relationen mit mehreren Interpretationsmöglichkeiten verarbeitet werden. Im zweiten Fall wird die Prämisse als Annotation in der artikulatorischen Schleife abgelegt. Das RO der Annotation wird dabei als *Anker* des LO bezeichnet. Das LO darf sich bezüglich der Relation höchstens adjazent zu seinem Anker befinden, nicht jedoch darüber hinaus, da sonst die Relation der Annotation nicht mehr gültig wäre. Diese Objekte sind also innerhalb des Modells in einem gewissen Rahmen variabel. Durch Vertauschungen dieser variablen Objekte werden alternative Modelle erzeugt, die mit dem ursprünglichen Prämissensatz konsistent sind. In der Variation wird nun versucht, ein Modell zu erzeugen, das der gegebenen Konklusion widerspricht. Ist dies nicht möglich, so ist die Konklusion in allen Modellen gültig.

Die annotierten Objekte können bis zu ihrem Anker bewegt werden. Ist bis dahin kein Gegenbeispiel generiert, wird versucht, den Anker nach dem selben Prinzip zu bewegen. Die rekursive Vertauschung setzt sich so lange fort, bis entweder ein Gegenbeispiel zur Konklusion gefunden wird, oder keine weiteren Vertauschungen mehr möglich sind. Die Transformationen erfolgen entsprechend den Bewegungsrichtungen der Basisrelationen (Abb. 4).

Komplexe Relationen werden für die Variation wieder in Basisrelationen zerlegt, die wie im bisherigen SRM variiert werden (vgl. [8]). Für Konklusionen mit komplexen Relationen wird das Modell zunächst inspiziert, um die im Modell gültige Konjunktion zu finden. Gilt keine der Konjunktionen, so ist das PMM selbst ein Gegenbeispiel.

Ist eine gültige Konjunktion gefunden, beginnt der eigentliche Variationsprozess. Die einzelnen Basisterme der Konjunktion werden als separate Konklusionen betrachtet und iterativ ein Modellvariationsprozess für jeden einzelnen Term durchgeführt. Nur wenn alle Terme der Konjunktion in allen möglichen Modellen gelten, gilt auch die komplexe Konklusion. Ein Gegenbeispiel ist also gefunden, falls für mindestens einen Basisterm die Variation zu einem Gegenmodell möglich ist.

Dadurch kann das Finden des Gegenbeispiels anhand von Konklusionen erklärt werden, die aus Basisrelationen bestehen. Ein Modell widerspricht dann einer Konklusion Rab , wenn es die konverse Relation $R^{-1}ab$ erfüllt. Dies bedeutet, dass entweder das LO a bis über das RO b vertauscht werden kann, oder analog der umgekehrte Fall.

Bei komplexen Annotationen bleibt zu klären, welches der Objekte als Anker fungiert. Dieses Problem wurde dadurch gelöst, dass Objekte mehrere Anker besitzen können.

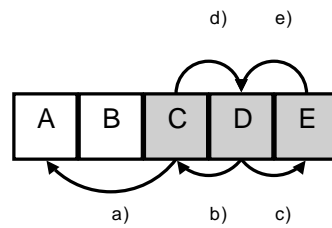


Abbildung 8: Eindimensionales Modell mit mehrfach annotierten Objekten. Die Pfeile zeigen von den annotierten Objekten zu deren Anker. Die Objekte C , D und E sind wegen der komplexen Annotation gesondert hervorgehoben.

Zur Verdeutlichung soll das PMM des Prämissensatzes

$$\begin{aligned} &A \text{ links } B \\ &A \text{ links } C \\ &D \text{ zwischen } C \text{ und } E \end{aligned}$$

betrachtet werden (siehe Abb. 8).

In diesem Modell werden die beiden Annotationen C rechts A und D zwischen C und E gespeichert. Diese halten die indeterminierten Positionen der Objekte C , D und E fest. Die Basisterme der komplexen Annotation und deren konverse Varianten werden bei der Verarbeitung als zusätzliche Annotationen betrachtet. Dies führt dazu, dass einige Objekte bezüglich mehrerer Richtungen annotiert sind, und somit Anker für verschiedene Richtungen besitzen. Objekt D etwa besitzt die beiden Anker C , als Grenze für die linke Position, und E als Grenze für die rechte Position.

3 Implementation

3.1 Software-Architektur

Das SRM wurde in der objektorientierten Programmiersprache Python implementiert und mit einer graphischen Oberfläche (Qt3) versehen.

Abbildung 11 zeigt einen Screenshot des Programms. Im linken oberen Bereich befindet sich der Eingabebereich für die Prämissen des Deduktionsproblems. Im aufgeklappten Auswahlmü sind die zur Verfügung stehenden Relationen zu sehen. Die Definitionen der Relationen werden direkt nach dem Programmstart aus den XML-Dateien ausgelesen und ins System integriert. In diesem Beispiel sind neben den vier Basisrelationen auch die drei komplexen Relationen *beside* (neben), *between* (zwischen) und *fleft* (far left) zu sehen.

Im mittleren Bereich der Oberfläche befindet sich ein grafisches Ausgabefenster, das die Modelle darstellt und die Fokusoperationen auf dem Grid animiert. Der Fokus ist hier durch eine rote Umrandung der entsprechenden Zelle hervorgehoben.

Im rechten oberen Bereich werden die Details zur Kostenanalyse der Fokusoperationen aufgelistet und bezüglich der einzelnen Deduktionsphasen aufgeteilt.

Links unten werden die Inhalte der Phonologischen Schleife (Verbal Memory) angezeigt. Zusätzlich zu den aktuell gehaltenen Annotationen werden auch die gelöschten und alle bisher gemachten Annotationen ausgegeben.

Unterhalb des grafischen Ausgabefensters befindet sich das Ausgabemodul der SRM-Deduktion. Neben der vorhergesagten Antwort wird auch die geschätzte Reaktionszeit eines menschlichen Probanden berechnet. Diese ergibt sich aus der Anzahl der nötigen Fokusoperationen und einer festgelegten Zeit für eine Operation. Standardmäßig wird mit 50ms pro Fokusoperation gerechnet. Des Weiteren wird die vorausberechnete Antwort bezüglich der Korrektheit evaluiert, in dem mittels einem Modellvariationsschritt, die Konklusion auf allgemeine Gültigkeit überprüft wird.

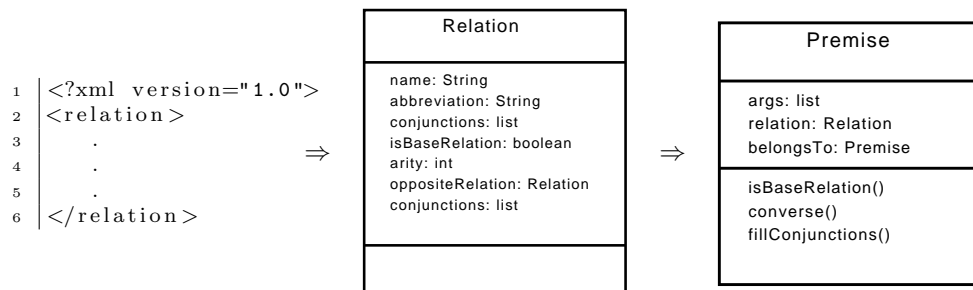


Abbildung 9: Interne Repräsentation der Relationen und Prämissen.

Intern besteht die Modellrepräsentation hauptsächlich aus den vier Klassen *ModelTree*, *Model*, *Layer* und *Focus*, die ineinander geschachtelt sind (siehe Abb. 10). Die *ModelTree*-Klasse ist eine Baum-Struktur, die sämtliche konstruierten Modelle enthält. Jedes Modell wiederum besteht aus mindestens einem Layer. Die Layer repräsentieren jeweils eine Gridstruktur, in der die Objekte platziert sind. Zu jedem dieser Layer gehört ein *Focus*-Objekt, das die Bewegungs- und Einfügeoperationen auf dem zugehörigen Layer ausführt.

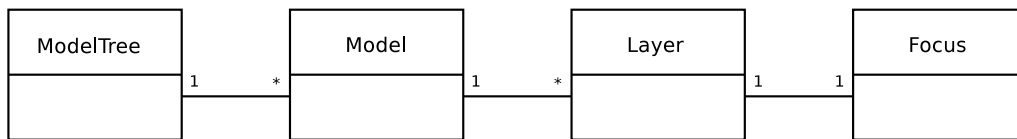


Abbildung 10: Assoziationen der Hauptklassen für die interne Modellrepräsentation.

Um die Verarbeitungsschritte detailliert festzuhalten und Kostenanalysen des Deduktionsprozesses zu erstellen, werden sämtliche Fokusoperationen abhängig von der aktuellen Deduktionsphase gezählt. Dazu senden die Fokusobjekte während der Verarbeitung Signale an das zugehörige Model-Objekt zurück. Die Kosten für Bewegungs-, Einfüge- und Leseoperationen lassen sich optional für jede der drei Deduktionsphasen anpassen.

Die in XML definierten Relationen werden nach dem Programmstart ausgelesen und als Relationsobjekte gespeichert. Diese wiederum sind wesentlicher Bestandteil der Prämissenobjekte. Die Relationen der Prämissen verknüpfen deren Argumente, die als Liste gehalten werden (siehe Abb. 9). Innerhalb der Prämissen werden die Argumente erst durch die Methode `fillConjunctions` mit der Relation verbunden.

Für die Modellierung der Phonologischen Schleife wurde die Listenstruktur *WorkingMemory* definiert. Die Länge der Liste kann optional festgelegt werden, um ein beschränktes Arbeitsgedächtnis zu simulieren. Die Klasse kann beliebig erweitert werden, so dass verschiedenste Speicherverdrängungsstrategien angewendet werden können. Implementiert wurden die beiden Strategien *FIFO* (First-In-First-Out) und *LRU* (Least Recently Used).

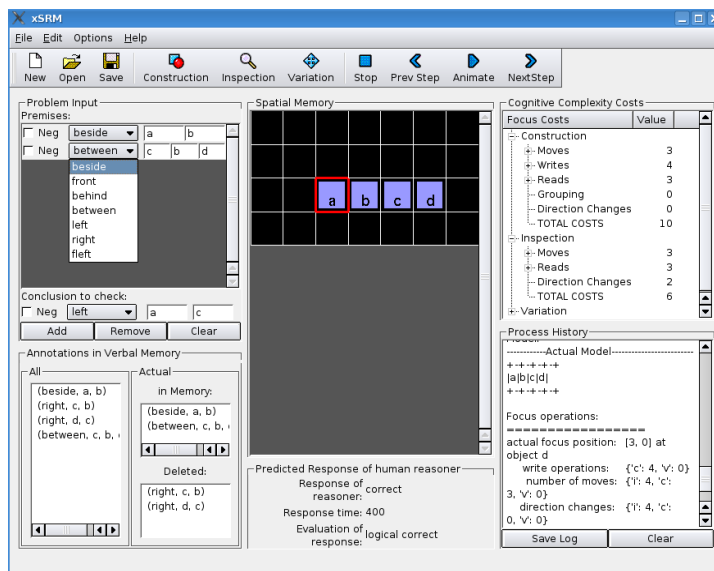


Abbildung 11: Screenshot der SRM-Implementation

3.2 Konstruktion

Der Konstruktionsalgorithmus erzeugt aus einer gegebenen Prämissenmenge einen Modellbaum, dessen linkstes Blatt dem PMM entspricht. Die Modelle der anderen Blätter enthalten konstruierte Modelle für die jeweiligen Interpretationen der verwendeten komplexen Prämissen.

Der erweiterte Algorithmus beinhaltet Routinen für die Relationszerlegung in Basisrelationen und die Integration der Prämissen in die Sub-Modelle des Modellbaums. Der ursprüngliche Konstruktionsalgorithmus wird für die Verarbeitung von Basisrelationen wiederverwendet.

```

1 | construct(premises, tree):
2 |     for p in premises:
3 |         if p.isBaseRelation():
4 |             integrate(p, tree)
5 |         else:
6 |             createModelBranches(p, tree)

```

Abbildung 12: construct-Funktion

Die Modellkonstruktion startet mit dem Aufruf der `construct`-Funktion (Abb. 12). Diese bekommt als Parameter die Prämissenmenge und ein zu diesem Zeitpunkt leeres `ModelTree`. Die Prämissen werden sukzessiv verarbeitet. Prämissen mit Basisrelationen werden direkt in den Modellbaum integriert. Für komplexe Prämissen werden zusätzliche Tochterknoten im Modellbaum erzeugt. Dazu werden die Prämissen an die Funktion `createModelBranches` weitergeleitet.

Sind die Prämissen so weit zerlegt, dass sie nur noch aus Basisrelationen bestehen, können sie mittels der `integrate`-Funktion (Abb. 13) in ein Modell integriert werden. Der Parameter `node` entspricht dabei dem Knoten des Modellbaums, in dessen Modell die Prämisse integriert werden soll. Die Angabe des Modellknotens ist für die Verarbeitung von zerlegten komplexen Prämissen nötig. Diese werden nur in das Modell integriert, in dem die aktuelle Interpretation der Relation festgehalten wird.

```

1 | integrate(premise, tree, node="All"):
2 |     if node == "All":
3 |         if tree.isEmpty():
4 |             model = createNode(tree)
5 |             model.baseConstruct(premise)
6 |         else:
7 |             for l in tree.leafs():
8 |                 l.BaseConstruct(premise)
9 |         else:
10 |             node.baseConstruct(premise)
11 |
12 | integrateTerms(tree, node, terms):
13 |     for t in terms:
14 |         integrate(t, tree, node)

```

Abbildung 13: Funktionen für die Integration der Objekte in die Knoten des Modellbaums.

Basisrelationen werden per Default (`node="A11"`) in die Modelle aller Blätter integriert. Ist der Baum leer, wird ein neuer Knoten mit einem leeren Modell erzeugt und die Prämisse dort eingefügt. Das eigentliche Einfügen in das Modell erledigt die Methode `baseConstruct` der Model-Klasse. Der dort verwendete Algorithmus entspricht dem bisherigen Konstruktionsalgorithmus des SRM (vgl. [8]).

Komplexe Prämissen werden in der Funktion `constructModelBranches` verarbeitet (Abb. 14). In dieser Funktion wird die Relation zuerst in die einzelnen Konjunktionen, bestehend aus Termen mit Basisrelationen, zerlegt. Für jede dieser Konjunktionen (Termmengen) wird ein neuer Tochterknoten an jedes Blatt des Modellbaums angefügt und die Terme der Konjunktion dort integriert. Ist der Baum leer, so werden die Tochtermodelle an den leeren Wurzelknoten gehängt. Das linkeste Modell im Baum entspricht der präferierten Interpretation der Relation und wird dadurch zum neuen PMM, dessen Fokusoperationen angezeigt und gezählt werden.

```

1 createModelBranches(premise, tree):
2     if tree.isEmpty():
3         model = createNode(tree)
4
5         for c in prem.conjunctions():
6             integrateTerms(tree, model, c)
7     else:
8         for l in tree.leafs():
9             for c in premise.conjunctions():
10                model = createNode(tree, l)
11                try:
12                    integrateTerms(tree, model, c)
13                except PremiseContradiction:
14                    tree.removeNode(model)

```

Abbildung 14: Pseudocode der Funktion `constructModelBranches`.

Bei der Konstruktion ist zu beachten, dass nicht notwendigerweise alle Konjunktionen konsistent in das Modell integriert werden können. Beispielsweise kann bei den Prämissen *A links B*, *B zwischen A und C* nur das lineare Modell *ABC* konstruiert werden, so dass alle anderen Interpretationen entfallen. Diese Fälle werden im letzten Abschnitt der Funktion geprüft und neu erstellte Knoten werden nur dann beibehalten, wenn die Terme der Konjunktion erfolgreich in das Modell eingefügt werden konnten.

Das Konstruktionsverfahren soll hier an einem Beispiel gezeigt werden. Der dazu verwendete Prämissensatz

B ist zwischen A und C
 D ist vor A
 E ist vor C
 A hinter F
 G ist zwischen F und H.

beschreibt ein ringförmiges Modell mit 8 Objekten. Des Weiteren existieren aufgrund der beiden *zwischen*-Relationen sehr viele weitere mögliche Objektanordnungen, wie der Modellbaum in Abb. 15 zeigt. Die Zwischenschritte des SRM der Modellkonstruktion sind in Abb. 16 dargestellt.

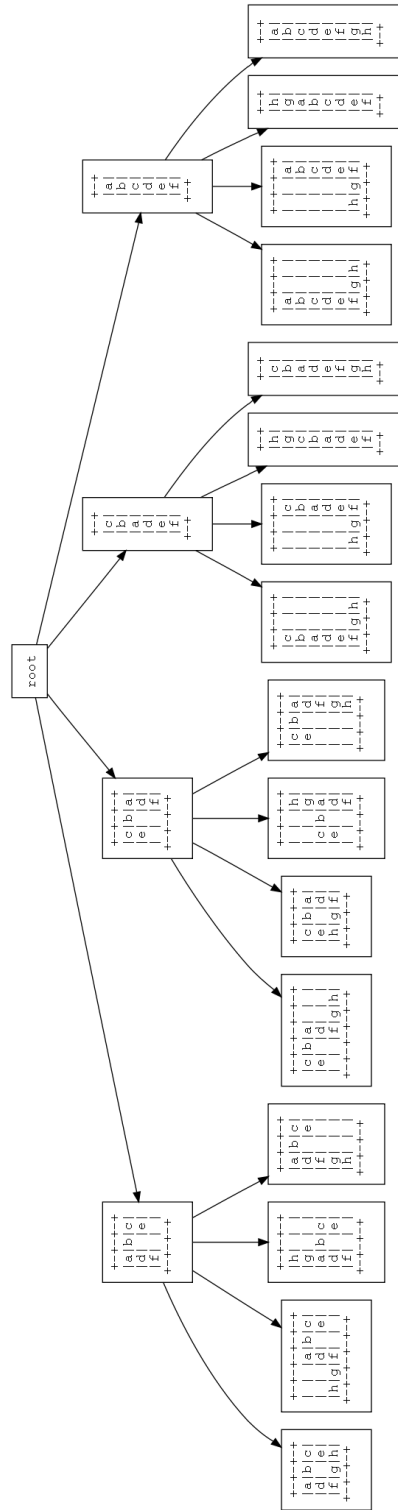


Abbildung 15: Modellbaum eines Prämissensatzes mit zwei *zwischen*-Prämissen. Ganz links befindet sich das ringförmige PMM.

Die initiale *zwischen*-Prämisse wird während der Verarbeitung in die Basisrelationen der entsprechenden Interpretationen zerlegt. Die daraus resultierende präferierte horizontale Interpretation besteht aus den Basisprämissen:

A ist links von B
B ist links von C
A ist links von C.

Diese werden nacheinander in das PMM integriert. Zuerst wird das Objekt *A* ins Modell eingefügt. Danach bewegt sich der Fokus ein Feld nach rechts (*conv(links)*) und fügt an diese Stelle Objekt *B* ein. Bei der zweiten Prämisse wird der Fokus nochmals ein Feld nach rechts bewegt und Objekt *C* an diese Stelle eingefügt. Die letzte Prämisse erfordert in diesem Fall einen *belief-revision*-Schritt, da nun alle Objekte der ursprünglichen *zwischen*-Prämisse im Modell enthalten sind. In diesem Fall wird bei ihrer Verarbeitung die Gültigkeit der Prämisse im Modell geprüft (Inspektion). Sollte die Inspektion ein negatives Ergebnis liefern, wird ein Modellvariationsschritt durchgeführt, um ein konsistentes Modell für die Prämissen zu finden.

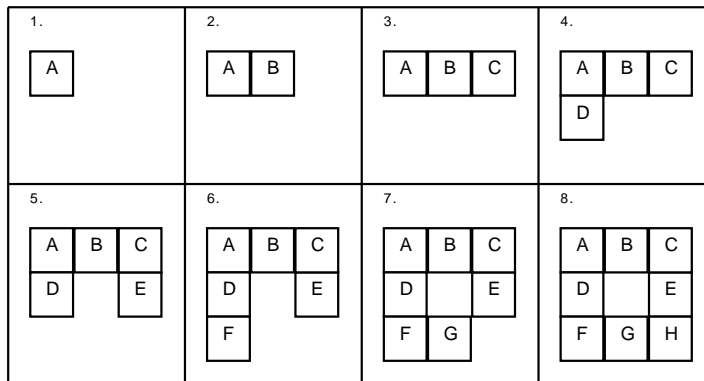


Abbildung 16: Sequentieller Ablauf der Modellkonstruktion. Die Schritte 1-3 und 7-8 sind Verarbeitungsschritte der Relation *zwischen*.

Für die verbleibenden möglichen Interpretationen werden jeweils neue Knoten im Modellbaum angelegt. Das vorläufige PMM, sowie die Modelle der alternativen Interpretationen, befinden sich im Modellbaum in den Knoten der Tiefe 1 (siehe Abb. 15). Zusätzlich zu den Objekten der initialen *zwischen*-Prämisse befinden sich in diesen Modellen auch die Objekte der drei folgenden Basisprämissen, da diese direkt in alle Blätter des derzeitigen Modellbaums integriert wurden. Für die zweite Prämisse, *D ist vor A*, bewegt sich der Fokus zuerst zurück auf das schon im Modell enthaltene Objekt *A*. Von dort aus bewegt sich der Fokus, der Relation entsprechend, nach „unten“ (vorn) und fügt Objekt *D* an der nächsten freien Position ein. Analog wird bei den beiden folgenden Basisprämissen vorgegangen. Bei der letzten Prämisse ist wieder eine Zerlegung in Basisrelationen nötig. Die Basisrelationen der präferierten Interpretation lauten hier:

F ist links von G
G ist links von H
F ist links von H

In diesem Fall ist das Objekt F schon im Modell enthalten. Die präferierte Interpretation erzeugt auch hier wieder eine horizontale Anordnung, wobei F am weitesten links platziert ist.

3.3 Inspektion

In der Inspektionsphase wird eine gegebene Prämisse auf ihre Gültigkeit im PMM geprüft. Der Algorithmus (siehe Abb. 17) ermittelt zuerst den Typ der eingegebenen Prämisse.

Komplexe Relationen werden auch hier vor der eigentlichen Verarbeitung in ihre Basisrelationen zerlegt, die dann einzeln untersucht werden. Für die Gültigkeit einer komplexen Relation werden die möglichen Interpretationen sukzessive geprüft. Eine Interpretation ist gültig, falls alle Terme der Konjunktion im Modell gelten. Dieser Test findet in der Funktion `complexInspect` (siehe Abb. 18) statt.

Für die Inspektion von Basisrelationen werden zunächst die Layer des LO und des RO ermittelt. Sind diese verschieden, oder Objekte der Prämisse nicht im Modell enthalten, so gilt die Relation nicht im Modell und die Funktion liefert `False` zurück. Anderenfalls wird der Fokus auf das RO der Relation bewegt und von dort aus das LO gesucht.

Um den Fokus auf dem kürzesten Weg durch das Modell und insbesondere über die im Modell enthaltenen Objekte zu leiten, wird mittels des Dijkstra-Algorithmus der kürzeste Weg berechnet. Dazu wird ein Graph aus den adjazenten Objekten des Layer berechnet (`buildGraph`). Die Fokusbewegung entlang des berechneten Pfades erlaubt es, die Fokusoperationen genau zählen und darstellen zu können.

```

1 inspect (prem, model):
2   if not prem.isBaseRelation():
3     return complexInspect (prem, model)
4   else:
5     ro_layer, lo_layer = model.layer (prem.RO), model.layer (prem.LO)
6     if ro_layer <> lo_layer:
7       return False
8     graph = ro_layer.buildGraph()
9     path = g.shortestPath (ro_layer.focus.pos, prem.RO)
10    ro_layer.focus.moveByPath (path)
11    path = g.shortestPath (ro_layer.focus.pos, prem.LO)
12    ro_layer.focus.moveByPath (path)
13    # simple coordinate check
14    return checkRelation (prem.LO, prem.RO, prem.relation)

```

Abbildung 17: Pseudocode des Inspektionsalgorithmus.

Die Erstellung des Graphen und die Berechnung des kürzesten Weges gehen nicht in die kognitiven Kosten der Inspektion ein. Diese berechnen sich ausschließlich aus den Fokusoperationen der Objektsuche, d.h. den Fokusoperationen beim Abschreiten des berechneten Pfades. Das Ergebnis der Inspektion wird über einen Koordinatenvergleich bestimmt (`checkRelation`). Dazu werden die Basisrelationen wie in Abb. 4 interpretiert. Für die Relation *links* beispielsweise gilt:

$$links(a, b) \Leftrightarrow pos_x(a) < pos_x(b) \wedge pos_y(a) = pos_y(b)$$

Die Funktionsweise des Inspektionsalgorithmus soll an dem oben konstruierten Beispiel gezeigt werden. Untersucht werden soll die komplexe Prämisse D zwischen A und F . Diese

```

1 | complexInspect(prem, model):
2 |   for c in prem.conjunctions:
3 |     res = True
4 |     for term in c:
5 |       res = res && inspect(term, model)
6 |       if not res: break
7 |     if res:
8 |       return True
9 |   return False

```

Abbildung 18: Pseudo-Code für Inspektion von komplexen Relationen.

taucht nicht in den Ausgangsprämissen auf, so dass sich das Ergebnis der Inspektion nur aus der im Modell enthaltenen Information ergibt. Alle Objekte befinden sich im Modell und im selben Layer, daher werden diese Tests im folgenden nicht erwähnt.

Am Ende der Konstruktion befindet sich der Fokus auf dem Objekt H in der rechten unteren Ecke des Modells. Da es sich um eine komplexe Prämisse handelt, wird gleich zu Beginn die Funktion `complexInspect` aufgerufen und die Konjunktionen der Relation sukzessive geprüft. Die Präferenzen der Relation sind in Abbildung 1 dargestellt. Zuerst werden die beiden horizontalen Interpretationen geprüft. Für die Prüfung der ersten Relation A links D wird der Fokus über den kürzesten Weg zum Objekt D , dem RO der Relation, gebracht (siehe Abb. 19 a)). Von dort aus wird das Objekt A gesucht, d.h. der Fokus dorthin bewegt und über den Koordinatenvergleich festgestellt, dass die Relation nicht gilt. Somit werden die Prüfungen der restlichen Terme abgebrochen und mit der zweiten horizontalen Interpretation weitergemacht. Da die drei Objekte vertikal angeordnet sind, verläuft auch diese Inspektion negativ. Als nächstes wird die erste vertikale Interpretation geprüft. Dazu werden die Basisterme

A hinter D
 D hinter F
 A hinter F

nacheinander inspiziert (siehe Abb. 19 b), c), d)). Alle drei Prämissen gelten im Modell, so dass die Inspektion jeweils `True` zurückliefert. Somit wurde eine erfüllende Konjunktion gefunden und die Inspektion der Ausgangsprämisse D zwischen A und F liefert `True` zurück.

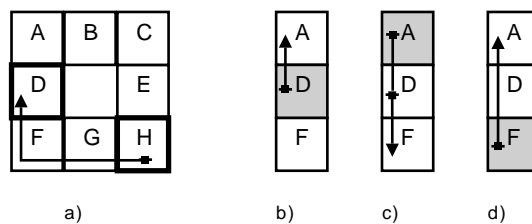


Abbildung 19: Fokusoperationen während der Modellinspektion. a) zeigt die Fokusbewegung von der letzten Fokusposition zum RO der ersten zu untersuchenden Prämisse. b), c) und d) zeigen die Fokusbewegungen beim testen der vertikalen Interpretation der *zwischen*-Prämisse.

3.4 Variation

Der Algorithmus der Modellvariation (Abb. 20) generiert über rekursive Vertauschungen alternative Modelle des PMM, um eine gegebene Konklusion auf Konsistenz zu prüfen (s.o.). Die Modellvariation ist nur im indeterminierten Modellfall, also bei Modellen mit mindestens einer Annotation, möglich. Nur annotierte Objekte können über Vertauschungsoperationen bewegt werden. Die nötigen Annotationen werden in der Listenstruktur der Phonologischen Schleife gespeichert. Für Objekte mit indeterminierten Positionen werden Annotationen in einer Listenstruktur gehalten. Nur annotierte Objekte können über Vertauschungsoperationen bewegt werden.

```

1 def complxVariation(conjunctions, model):
2     for c in conjunctions:
3         if not model.inspect c:
4             continue
5         else:
6             if model.variate_conj(c):
7                 break
8     else:
9         # model is counter-example
10        return False
11    # conclusion holds in all models
12    return True
13
14 def variate_conj(c):
15     for term in c:
16         res = model.variate(t)
17         if not res:
18             # counter-example
19             return False

```

Abbildung 20: Funktionen für die Variation komplexer Prämissen.

Zu Beginn der Variation (`complxVariation`) wird über alle Konjunktionen der Konklusion iteriert. Jede Konjunktion wird im Modell auf Gültigkeit inspiziert. Gilt sie nicht im Modell, wird der aktuelle Schleifendurchlauf übersprungen. Gilt keine der Konjunktionen, so ist das PMM selbst ein Gegenbeispiel für die Konklusion und die Variation ist beendet.

Ist eine gültige Konjunktion gefunden worden, beginnt die rekursive Vertauschung. Nun werden sämtliche Basisterme iterativ variiert, wobei jeder einzelne Term wie eine separate Konklusion behandelt wird.

Die Modellvariation soll hier an dem in Abschnitt 2.8 vorgestellten ein-dimensionalen Beispiel mit dem PMM aus Abb. 8 gezeigt werden.

Für die Modellvariation soll nun die Konklusion *D ist zwischen B und E* untersucht werden. Die erste Konjunktion, die geprüft wird, entspricht der horizontalen Interpretation der Konklusion. Die dazugehörigen Basisterme lauten

B links D
D links E
B links E.

Alle Basisterme gelten im PMM. Die Inspektion der Konjunktion, liefert also *True* zurück und die Funktion zur Variation der Basisprämissen (`varitate_conj`) wird aufgerufen.

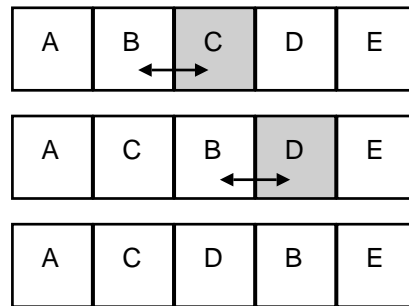


Abbildung 21: Rekursive Vertauschungsschritte während der Modellvariation.

Der Variationsalgorithmus prüft nun den ersten Basisterm im Sinne einer Konklusion. Es wird also zuerst untersucht, ob in allen Modellen die Relation *B links D* gilt. Dazu als erstes versucht, das Objekt *B* (LO) über das Objekt *D* hinaus zu tauschen. Da *B* nicht annotiert ist, kann es nicht direkt bewegt werden. Deswegen wird im zweiten Schritt der umgekehrte Weg probiert, und versucht, Objekt *D* bis zu seinem Anker oder über *B* hinaus zu verschieben. Das Objekt ist mehrfach annotiert, und der Anker für die Verschiebung nach links ist das adjazent liegende Objekt *C*. *D* kann also nicht mehr weiter nach links verschoben werden. Statt dessen wird nun im rekursiven Schritt versucht, das Ankerobjekt *C* nach links über das Objekt *B* zu bringen. Der Anker von *C*, das Objekt *A*, ist noch nicht erreicht, daher ist der Tausch mit dem Nachbarobjekt möglich, und *C* kann somit über *B* bewegt werden (vgl. Abb. 21). Daraus folgt, dass auch *D* über *B* hinaus bewegt werden kann und somit ein Gegenbeispiel für die Konklusion gefunden wurde.

4 Zusammenfassung und Diskussion

In dieser Studienarbeit wurde gezeigt, wie die Verarbeitung komplexer Relationen beim qualitativen räumlichen Schließen auf die Verarbeitung einfacher transitiver Relationen (L, R, F, B) zurückgeführt werden kann. Komplexe Relationen werden dazu in eine Beschreibung aus Basisrelationen zerlegt, die nach den bisher untersuchten Präferenzstrategien verarbeitet werden. Die Aufzählung sämtlicher Interpretationsmöglichkeiten, erlaubt es, auch sehr komplizierte Relationen zu beschreiben. Die Menge der möglichen definierbaren Relation wird nur durch die Tatsache eingeschränkt, dass die Repräsentation der Relationen durch transitive Basisrelationen beschreibbar sein muss (siehe Kapitel 2). Die Basisrelationen des SRM-Modells sind bisher nur für den zweidimensionalen Raum definiert. Relationen, die eine dreidimensionale Objektanordnung erfordern, können somit nicht beschrieben und verarbeitet werden. Da der Mensch natürlich dazu in der Lage ist, mit dreidimensionalen Objektanordnungen umzugehen, ist eine Erweiterung des SRM dahingehend wünschenswert.

Für die Verarbeitung der mehrstelligen und meist mehrdeutigen Relationen wurden die Algorithmen der einzelnen Deduktionsphasen erweitert und in die bestehende Implementation des SRM integriert. Besonders die Möglichkeit die im SRM zur Verfügung stehenden Relationen nun selbst definieren und erweitern zu können, bietet enorme Freiheiten für Experimentieranordnungen. Mit den Basisrelationen können so auch andere transitive Relationen (z.B. *größer als*) definiert werden, oder auf mehr als zwei Argumente erweitert werden, so daß Aussagen wie *A ist größer als B und C* verarbeitet werden können. Des Weiteren bietet sich auch die interessante Option, durch Änderungen an der Reihenfolge der Relationsinterpretationen den Deduktionsprozess mit verschiedenen Interpretationspräferenzen analysieren zu können.

Bisher ist das hier vorgestellte Verarbeitungskonzept allerdings ein rein theoretisches. Erste Untersuchungen deuten aber darauf hin, dass diese Form der Verarbeitung tatsächlich so stattfinden könnte. Für die exakte semantische Interpretation komplexer Relationen sind jedoch weitere empirische Untersuchungen notwendig. Insbesondere muss geklärt werden, welche Relationen als Basisrelationen in Frage kommen und direkt verarbeitet werden.

Bei der Modellkonstruktion wurde ein Modellbaum eingeführt, in dem für mehrdeutige Relationen die verschiedenen Interpretationen festgehalten werden. Dies war notwendig, da die diversen Interpretationen nicht über Annotationen realisiert werden konnten. Der Modellbaum spielt aus kognitiver Sicht für die Verarbeitung keine große Rolle, da, wie oben erwähnt, nicht erwartet werden kann, dass Menschen einen solchen Baum mental konstruieren können. Er bietet jedoch einen guten Überblick über mögliche Alternativmodelle eines Problems. Bei der Verarbeitung von vielen mehrdeutigen Relationen, enthält der Baum auch dementsprechend viele Verzweigungen. Widersprüchliche Interpretationen werden nicht in den Baum integriert. Durch die Übersicht über die möglichen Interpretationen der im Deduktionsproblem enthaltenen Relationen und kann so sehr gut nachvollzogen werden, bei welchen Repräsentationen und Relationsinterpretationen Versuchspersonen sehr wenige oder sehr viele Fehler machen.

Der genaue Ablauf der kognitiven Prozesse beim räumlichen Schließen ist selbst für die einfachen Basisrelationen nicht vollständig geklärt. Mittlerweile können jedoch zahlreiche experimentell festgestellte Effekte erklärt und in kognitiven Modellierungen wie dem SRM oder ACT-R nachvollzogen werden. Die hier vorgestellte Erweiterung des SRM versucht, die bekannten grundlegenden Prinzipien des relationalen Schließens zu bewahren. Es wird daher erwartet, dass die Vorhersagen, die mit dem SRM gemacht werden können, mit Ergebnissen menschlicher Versuchspersonen übereinstimmen. Zu diesen Prinzipien gehört das

Ökonomizitätsprinzip bei der Modellkonstruktion, das unter anderem besagt, dass die Prämisseninformationen sukzessiv in die Modelle integriert werden. Bei den bisher integrierten Relationen *neben* und *zwischen* wurde bei der Definition auch auf das Prinzip der *order-of-mention* geachtet. Die Präferenzen der Relationsinterpretationen können zwar selbst festgelegt werden, dabei sind jedoch empirisch untersuchte Präferenzen zu berücksichtigen.

Die vorgestellten Algorithmen sind nicht auf die Verwendung im SRM beschränkt, sondern können auch in anderen kognitiven Modellierungssystemen wie etwa ACT-R Anwendung finden. Weiterhin können die Algorithmen durch die objektorientierte Implementation leicht angepasst und erweitert werden, so dass neue Forschungsergebnisse einfach integriert werden können.

In dieser Arbeit wurden komplexe Relationen nur für die räumlich-relationale Domäne betrachtet. Für viele andere Domänen, wie z.B. RCC-8 würde dieser Ansatz so nicht funktionieren. Die Verarbeitung komplexer Relationen benötigt meist eine spezielle Anpassung an die jeweilige Domäne. Jedoch ist auch denkbar, temporale Relationen mit diesem Ansatz zu verarbeiten. Betrachtet man etwa die Intervallrelationen nach Allen, so läßt sich ein Intervall durch räumliche Entitäten eines Start- und eines Endpunkts darstellen. Eine temporale Anordnung der Intervalle käme dann einem räumlich-relationalen Anordnungsproblem gleich. Die Untersuchung und Modellierung kognitiver Strukturen, insbesondere mentaler Modelle, ist natürlich in erster Linie für Psychologen und Kognitionswissenschaftler von Interesse. Da die Verarbeitung von räumlichen Informationen aber auch für die Informatik von Nutzen sein kann und der Mensch scheinbar mühelos mit diesen Informationen umgeht, ist es sinnvoll, diese Prozesse so genau wie möglich zu untersuchen und nachzuvollziehen. Eine denkbare Anwendung mentaler Modelle sind z.B. Situationen von im Raum agierenden Agentensystemen, bei denen die genaue Position des Agenten nicht unmittelbar benötigt oder feststellbar ist. So kann mittels qualitativen räumlichen Beschreibungen und bestimmten Wegmarken eine ungefähre Übersicht der Umgebung, sozusagen eine qualitative Landkarte, erzeugt werden, so dass eine approximative Lokalisation möglich wird.

Literatur

- [1] EBBINGHAUS, H.-D., J. FLUM und W. THOMAS: *Einführung in die mathematische Logik*. BI Wissenschaftsverlag, 3. Auflage, 1992.
- [2] GOODWIN, G.P. und P.N. JOHNSON-LAIRD: *Reasoning About Relations*. Psychological Review, 112(2):468–93, 2005.
- [3] HENST, J.-B. VAN DER: *Mental model theory versus the inference rule approach in relational reasoning*. Thinking and Reasoning, 8(3):193–203, 2002.
- [4] HENST, J.-B. VAN DER und W. SCHAEKEN: *The Wording of Conclusions in Relational Reasoning*. Cognition, 97(1):1–22, 2005.
- [5] JAHN, G., P.N. JOHNSON-LAIRD und M. KNAUFF: *Reasoning About Consistency with Spatial Mental Models: Hidden and Obvious Indeterminacy in Spatial Descriptions*. In: FREKSA, C., M. KNAUFF, B. KRIEG-BRÜCKNER und B. NEBEL (Herausgeber): *Spatial Cognition IV: Reasoning, Action, Interaction*, Seiten 165–180. International Conference Spatial Cognition, Springer, 2004.
- [6] OTTMANN, T. und P. WIDMAYER: *Algorithmen und Datenstrukturen*. Spektrum Akademischer Verlag, 4. Auflage, 2002.
- [7] RAGNI, M., T. FANGMEIER, L. WEBBER und M. KNAUFF: *Preferred Mental Models: How and Why They Are So Important in Human Reasoning with Spatial Relations*. In: BARKOWSKY, T, M. KNAUFF, G. LIGOZAT und D.R. MONTELLO (Herausgeber): *Spatial Cognition V: Reasoning, Action, Interaction*, Seiten 175–190. International Conference Spatial Cognition, Springer, 2006.
- [8] RAGNI, M., M. KNAUFF und B. NEBEL: *A Computational Model for Spatial Reasoning with Mental Models*. In: BARA, B., B. BARSALOU und M. BUCCIARELLI (Herausgeber): *Proceedings of the 27th Annual Cognitive Science Conference (CogSci05)*, Seiten 1064–70. Erlbaum, 2005. ISBN 0-9768318-1-3.
- [9] RAUH, R., C. SCHLIEDER und M. KNAUFF: *Präferierte mentale Modelle beim räumlich-relationalen Schließen: Empirie und kognitive Modellierung*. Kognitionswissenschaft, 6:21–34, 1997.
- [10] SCHLEIPEN, S.: *Negation beim räumlichen Schließen*. Studienarbeit, Oktober 2007.
- [11] SCHLIEDER, C.: *The Construction of Preferred Mental Models in Reasoning with Interval Relations*, Kapitel 14, Seiten 333–357. Elsevier Science B.V., 1999.
- [12] SCIVOS, A. und B. NEBEL: *The Finest of Its Class: The Natural Point-Based Ternary Calculus LR for Qualitative Spatial Reasoning*. In: FREKSA, C., M. KNAUFF, B. KRIEG-BRÜCKNER, B. NEBEL und T. BARKOWSKY (Herausgeber): *Spatial Cognition IV: Reasoning, Action, Interaction*, Seiten 283–303. International Conference Spatial Cognition, Springer, 2004.