A Cognitive Computational Model for Spatial Reasoning

Marco Ragni and Felix Steffenhagen
University of Freiburg
Department of Computer Science
D-79110 Freiburg, Germany

Abstract

In recent years a lot of psychological research efforts have been made in analyzing human spatial reasoning. Psychologists have used implicitly many spatial cognitive models, i.e. a model of how humans conceptualize spatial information and reason about it, based on the mental model theory to model their experimental findings. But only little effort has been put into identifying from an algorithmic point of view the control mechanism in cognitive models for reasoning with spatial relations. Without having such a specification the task of testing and improving cognitive models seems to be rather difficult, whereas the transfer of such cognitive models with attention to AI systems seems to be even more important. Only a precise computational model defining parameters and operations make testable predictions. In this paper we extend the SRM model, by embedding it into Baddeleys Working Memory Model. By this embedding we can define the role of the central executive and show that this subsystem plays an important role in precising the cognitive attention.

Introduction

The ability to deal with spatial and temporal information is one of the most fundamental skills of any intelligent system and important in our everyday lives. When route descriptions are given, usually spatial information is contained in the description. While in engineering or physics it is most common to represent spatial information quantitatively, e.g. using coordinate systems, human communication mostly uses a qualitative description, which specifies qualitative relationships between spatial entities. But how is this information processed? Where is the focus of cognitive attention in processing qualitative information? In the following we concentrate on relational reasoning problems, e.g.

The apple is to the left of the lemon.
The apple is to the left of the orange.
The kiwi is to the right of the orange.

Is the kiwi (always) to the right of the lemon?

The statements are called *premises*, the fruits are the *terms*, and the question refers to a putative *conclusion*. A premise of the form “The apple is to the left of the lemon” consists of (two) objects (apple and lemon), and a (usually binary) relation like “to the left of”. More precisely, the first object (apple) is the ”to be localized object”(LO), which is placed according to the relation (left of) of the second object (lemon), which is the “reference object” (RO) (Miller & Johnson-Laird 1976). Such relational problem can be abbreviated by the tuple $(\mathcal{P}, \varphi)$, with premises $\mathcal{P}$ and a putative conclusion φ. There are basically two main cognitive, as well as mathematical, approaches about how humans solve such problems: syntactic-based theories and semantic-based theories. For example (Rips 1994) suggested that humans solve such tasks by applying formal transitivity rules to the premises, whereas in (Huttenlocher 1968) it is proposed that humans construct and inspect a spatial array that represents the state of affairs described in the premises. The first theory argues that human deduction can be compared to searching and finding mental proofs. Difficulty arises if a high number of rules must be applied to verify a conclusion. The other approach which has been further elaborated on in the mental models theory (MMT) of relational reasoning (Johnson-Laird & Byrne 1991) and (Johnson-Laird 2001) is generally the more accepted theoretical account in human reasoning in terms of empirical and neuronal evidence. According to the mental model theory, linguistic processes are only relevant to transfer the information from the premises into a spatial array and back again, but the reasoning process itself relies on model manipulation only. A *mental model* (in the usual logical sense) is a structure in which the premises are true. Psychologically, such a mental model is interpreted as an internal representation of objects and relations in spatial working memory that matches the state of affairs given in the premises of the reasoning task. The semantic theory of mental models is based on the mathematical definition of deduction, i.e. a propositional statement φ is a consequence of a set of premises $\mathcal{P}$, written $\mathcal{P} \models \varphi$, if in each model $\mathcal{A}$ of $\mathcal{P}$, the conclusion φ is true. The mental model theory describes the human reasoning process in three distinct phases (Johnson-Laird 2001). If new information is encountered during the reading of the premises, it is immediately used in the construction of the mental model, which is the so-called *generation phase*. Then in the *inspection phase* the model is inspected to check if the putative conclusion is consistent in the model at hand. Finally, in the *validation phase* alternative models are constructed from the premises that refute this putative conclusion. In our exam-

---

ple above, the spatial description is not fully specified, since in the first two premises “A is to the left of L” and “A is to the left of O” the exact relation in-between “O” and “L” is not specified. Such problems lead to multiple-model cases since both models A L O and A O L fulfill the premises. With the number of models which have to be handled simultaneously, the cognitive difficulty arises (Johnson-Laird 2001). The classical mental model theory is not able to explain a phenomenon encountered in multiple-model cases, namely that humans in general tend to construct a *preferred mental model*. This model is easier to construct, less complex, and easier to maintain in working memory compared to all other possible models (Knauff *et al.* 1998). In the model variation phase, this PMM is varied to find alternative interpretations of the premises (Rauh *et al.* 2005). However, from a formal point of view, the mental model theory has been mostly used in a rather informal way and is not fully specified. Also no operations or manipulations are described. In other words, the use, construction, and inspection of mental models have been handled in a rather implicit and vague way (Johnson-Laird 2001; Baguley & Payne 1999; Vandierendonck, Dierckx, & Vooght 2004). A first approach in precising a computational model for the preferred mental model theory has been presented in (Ragni, Knauff, & Nebel 2005). This model consists of an input device for the premises, a two-dimensional spatial array where the mental model is constructed, inspected, and varied, and a focus which performs these operations. This cognitive model was able to explain many cognitive results in the literature, e.g. (Knauff *et al.* 1998) by applying a standard cost measure for each necessary model operation. Future work tested predictions made by the models empirically (Ragni *et al.* 2006).

Without having an algorithmic formalization of a cognitive model, the task of testing and improving this model seems to be rather difficult, whereas the transfer of such cognitive models to AI systems seems to be even harder. Only a precise computational model, defining parameters and operations, makes testable predictions. Furthermore, by formally precising the role of the subsystems of a cognitive model, i.e. its store systems and by having empirical datas at hand, it is possible to identify the needed abilities a computational model must provide.

In this paper we work out the main assumptions found in the literature of human spatial (relational) reasoning (Section 2) and present a formalization of this in a computational model (Section 3). Through this formalization and empirical evidence, we aim at identifying the role of control mechanisms algorithmically in Baddeleys Working Memory Model. Finally, we give a short discussion of our results.

Psychological Background

The computational model proposed in this paper is based on Baddeleys’ Working Memory Model (BWMM) from (Baddeley 1999) and the theory of preferred mental models. The WMM assumes a central executive, which is responsible for monitoring and coordinating the operations of two subsystems, the *phonological loop* (PL) and the *visuo-spatial scratchpad* (VSSP) (Figure 1). The first subsystem, the PL, allows to store information in a language-based form. Another subsystem, the VSSP, independent from the PL in terms of limits, stores visual and spatial information. Both subsystems are controlled by a central executive which is able to store and manipulate information in both subsystems. For combining the PMMT and the BWMM, the following questions have to be answered: In which subsystem and how takes the reasoning place? Which limits do the subsystems and the control process have? Which control mechanisms exist for the different subsystems? These questions are answered by results from the literature: The deduction process in relational spatial reasoning uses mental models (Byrne & Johnson-Laird 1989), which can be located in the WMM in the visuo-spatial sketchpad (Huttenlocher 1968; Vandierendonck, Dierckx, & Vooght 2004), where the mental models are constructed and manipulated as well. The model in the VSSP is manipulated by a special device which is called focus (Vandierendonck, Dierckx, & Vooght 2004; Baguley & Payne 1999). The phonological loop uses some dynamic memory allocation like the first-in-first-out principle (Baguley & Payne 1999).

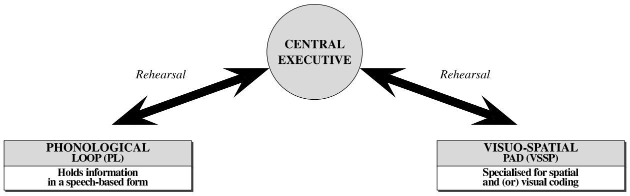


Figure 1: Baddeleys (1999) Working Memory Model.

Since the existence of preferred mental models is widely accepted (Knauff *et al.* 1998; Rauh *et al.* 2005; Johnson-Laird 2001), and follows the principle of economy (Manktelow 1999), we have to identify (and model) strategies of how the computational model should deal with indefinite information. The preferred mental model is constructed out of the given premises using such strategies to specify the movements of the focus. In case of indeterminacy, i.e. the construction of more than one model is possible, humans tend to construct a specific model, which we refer to as preferred mental model. The principle of economicity is the determining factor in explaining human preferences (Manktelow 1999). It also explains that a model is constructed incrementally from its premises. Such a model construction process saves working memory capacities because each bit of information is immediately processed and integrated into the model (Johnson-Laird & Byrne 1991). In other words, the information of the premises does not have to be stored, i.e. the information of a new premise is immediately integrated in the model.

The $\mathcal{CROS}$ Model

Each computational model is based on assumptions and abstractions depending on its aim. The $\mathcal{CROS}$-Model (Cognitive Relational Operating Systems) formalizing the WMM and PMMT consists of: A conceptualization of the WMM (with subsystems), a *manipulation device* for the mental models, a (relational) language describing object positions, and a *semantic interpreter* interpreting the language. The central place where models are located is the visuo-spatial scratchpad. The VSSP is a spatial array (SA) of two-dimensional grids, called *layer*, where the models are generated and manipulated by a device called *focus*. The focus can perform a small number of operations like moves, reads, and inserts. Multiple layer can exist in model descriptions where two subsets of objects cannot be logically connected, so that no conclusion about their relation to each other can be drawn.

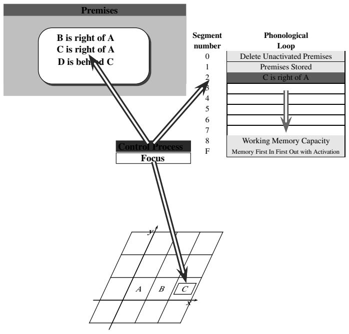


Figure 2: The $\mathcal{CROS}$-Model.

For example for ’A left B’ and ’C right D’, there are two possible submodels, each placed in its own layer, so that submodel AB would be in the first and CD in the second layer.

Problems related to the ambiguity of spatial relations are not accounted - the model interprets the string “A is left of B” as both objects that are in the same line, and A is to the left of B. The relations “right”, “front”, and “behind” are equivalently defined. When processing natural language strings, the meaning of the input has to be interpreted. In linguistics, as well as in psychology, the existence of a *semantic interpreter* (SI) is assumed, which in our model maps syntactically analyzed texts to the formal representation. The semantic interpretation is not part of this paper, so we simply assume a parser that provides the correct meanings to the system. More complex relations like ’in-between’ or negated relations can be formulated as small algorithms defining these relations in terms of focus movements and base relations. This allows the integration of the relational complexity approach from Halford (Halford, Wilson, & Phillips 1998) and the modeling of decomposability (Goodwin & Johnson-Laird 2005).

If, as in the example above, indeterminacy occurs, information about other possible models must be stored. Since a mental model is only a representation, i.e. it is one model, the information of other models must be hold in another subsystem. This information is psychologically modeled via *annotations* on objects (Vandierendonck, Dierckx, & Vooght 2004). Since we do not know how indeterminate information is encoded in human mind, we use the complete premise as annotation. The appropriate memory system in the WMM for this kind of propositional information is the phonological loop. This goes along with neurological evidence (Knauff *et al.* 2002). The PL is managed by a dynamic memory allocation system like FiFo or least-recently-used strategy (LRU) - this allows the modeling of activated objects.

Since both systems, the SA and the PL, are only memory systems and the focus manipulates only the SA, a *control process*, the “program” of the $\mathcal{CROS}$ is needed, managing the subsystems and controlling the focus operations on the SA. The control process has a limited instruction set (Table 1). Several instructions directly control the read/insert/move operations of the focus, statements to branch or loop the control flow and simple test instructions. With this set of instructions, algorithms for all three deduction phases can be defined and different insertion strategies can be tested (and compared). The premises are iteratively read and interpreted by the SI, and the control process immediately inserts the new encountered information into the model by steering the focus on the SA and adding indeterminacy information to the PL. For premises that cannot be constructed into one layer the focus has the ability to create new layers. Formally, the main parts of a $\mathcal{CROS}$ system are:

- I: the input device
- SI: the semantic interpreter mapping the syntactical input of I to propositional form.
- A: a spatial array containing the layers. We define $\omega(A)$ as the objects held by the array A and $\lambda(O)$ as the layer of object O.
- F: the focus that works on spatial array. It can perform move operations (L,R,B,F,No-Move) as well as grouping and shift operations.
- PL: the PL, a memory system to store propositional information.
- C: a control process using the instructions defined in Table 1 that is responsible for controlling all subsystems.

In the following we present the algorithms for the construction, inspection and variation, for the initially presented problem, abbreviating the fruit objects with the initial letters:

A is to the left of L
A is to the left of O
K is to the right of O

Model construction The algorithm for the model construction has to distinguish five types of premises (O_1, r, O_2) to place the objects of the premises: (1) $\omega(A) = \emptyset$ (first premise), (2) $O_1 \in \omega(A)$ and $O_2 \notin \omega(A)$ or vice versa, (3) $O_1, O_2 \notin \omega(A)$, (4) $\lambda(O_1) \neq \lambda(O_2)$ (connecting two layers), (5) $\lambda(O_1) = \lambda(O_2)$ (additional knowledge).

The construction process begins with the first premise and an empty layer, first placing the RO, then moving in the direction of the relation where the LO is placed in the next free cell. In the example, L is inserted first, the focus moves to the left and inserts A. The algorithm (Figure 3) checks the type of each new premise and inserts the object(s) according to the specific case. For premises of type 2 only one object will be inserted into the model according to the already contained object. If the new object cannot be placed as a direct neighbor, the model structure is indeterminate, and the control process annotates the object by inserting the relational information as a proposition into the PL, and the focus places the object at hand according to the fff-principle. For premises of type 3, where none of both objects are contained in the model, a new layer is generated, and both objects will be placed as in the beginning of the model construction. If both objects are contained in different layers (type 4), both layers have to be merged according to the relation of the premise. Premises of type 5 specify additional knowledge for two objects contained in the same layer. They are processed by a model variation step, trying to check if the inverse premise holds in all variations of the actual model. If a counter-example exists, it is a model containing the additional knowledge. The second premise is of type 2, because A is already in the model, so O is inserted to the right of L according to the fff-principle, and gets the annotation 'RA', meaning that its position is always to the right of A. The next processed premise is also of type 2 and object K, that is not in the model, is inserted directly to the right of O. But because O has an annotation, K has to be annotated too. Now the construction phase is complete, the constructed model is shown in the first line of Figure 6.

Control process operations	
readnext()	read the next premise from SI and save values to the variables LO, RO, and REL
SubSystem(sys)	change sub system the central process is working on; sys can be the PL or the SA
Control Flow	
if val then {instr. block} [else {instr. block }]	test whether val is true and process first instruction block else 2nd block is processed
while val do <instr. block>	process instr. block as long as $val <> 0$
Operations on Phonological Loop	
Command	Description
writep(prem)	write premise into loop
annotate(o,a)	annotate object o with a
annotations(o)	return annotations of objects o
annotated(o)	return true if o is annotated
Focus Operations	
Command	Description
fmove(d)	move the focus to direction d
fread()	read cell where focus is on, return false if cell is empty
fwrite(o)	write object o to cell where the focus is on
newLayer()	create new empty layer
Complex Sub-Programs	
sub program	description
shift(o, d)	shift object o to direction of d
exchange(o,rel,concl)	exchange object o the direction of rel generating a new model
fmoveto(o)	move focus to object o
inverse(rel)	compute inverse relation to rel
layer(o)	returns the layer of object o, false if o is not in any layer
merge(l_1, l_2)	merge layer l_1 and l_2

Table 1: The instruction set of the $\mathcal{CROS}$.

```
def constructModel():
 readnext()
 fwrite(RO)
 fmove(inverse(REL))
 fwrite(LO)
 while readnext() do
 { if type2 then
   { fmove focus to contained obj
       fmove focus
       while not placed do
       {  if fread() then
            annotate missing obj
         else
            fwrite missing object
            placed = true} }
   if type3 then
   { l = newLayer()
     fwrite(LO); fmove(inverse(REL))
     fwrite(RO) }
   if type4 then
     merge(layer(LO), layer(RO))
   if type5 then
   { newModel=valConcl(LO,inverse(REL),RO)
     if newModel then
       writeModel() }}
```

Figure 3: The construction algorithm.

Model inspection After model construction, the *inspection phase* checks the putative conclusion (cp. Figure 4). The focus moves to the first given object (RO), and from there it inspects the model according to the relation in order to find the second object (LO). The search process terminates since the model is bounded by its number of objects n, so no more than $O(n)$ steps are necessary.

```
focus_inspect(LO,rel, RO):
  fmoveto(RO)
    while fread != LO do
    {    fmove(rel)
         if fread()==LO then
           return true }
  return false
```

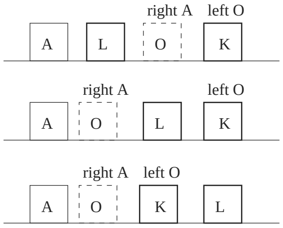
Figure 4: Pseudo code for the inspection algorithm.

Model variation The *model variation* comes into play if a conclusion must be verified or if additional knowledge of two already contained objects must be processed during the model construction process. The focus starts in the variation process with the PMM and variates it with local transformations to generate a counter-example to the putative conclusion at hand.

The variation process starts from the generated PMM (in which the putative conclusion holds). The algorithm checks whether one of the objects in the conclusion is annotated. An annotation on an object specifies the positional relation to reference objects, we refer to as *anchor*. If the annotations on one of the objects include the relation and the other object of the putative conclusion then the putative conclusion holds. The same argument holds if none of the conclusions' objects appear in the annotations because the positions of the objects are then determined. If there is an annotation on one object (and not to the other), as in the example conclusion 'K is to the right of L' (see Figure 6), the only object of the conclusion to be moved is K and not L. This goes along with the use of annotations, i.e. in the construction process an annotation is made only for indeterminate object positions. If the object to be moved has an anchor, it may be necessary to move the anchor first. To provide an example: K cannot be moved, because O, the anchor of K, is a direct neighbor of K. Thus, the algorithm first exchanges the anchor to the left of L, which is possible since A is the anchor of O. Now the counter-example can be generated by exchanging K and L, because the anchor of K can be placed to the left of L, so false is returned. If both objects are annotated, then first the LO of the putative conclusion is exchanged. LO is exchanged into the direction of RO until its anchor is reached. If thereby an inconsistent model is generated, the algorithm stops and returns false. It is possible that the anchor object lies between LO and RO, so LO is exchanged until it reaches the anchor. Then the anchor object is recursivly exchanged towards the RO. If there no further exchanges to RO are possible, the exchange process starts to exchange the RO into the direction of LO.

```
validateConclusion(Model, concl):
{if layer(LO) != layer(RO)
    return false
  if not check(concl)
    return false
  if conclusion in annotations(Model)
   or inverse(concl) in annotations(Model)
    return true
  if LO not in objects(annotations)
   or RO  not in objects(annotations)
    return true
  if not exchange(RO, relation, concl)
    return false
  else
    if not exchange(LO, rev(relation), concl)
      return false
  return true }
```

Figure 5: Model variation algorithm. The exchange method exchanges an object according to the given premise and conclusion to find a counter-example. The object is exchanged until the 'anchor' object is reached, from there it recursively proceeds with the anchor and so on.



The first line shows the model after construction according to the premises. The bold marked objects shall be variated to check a conclusion. The dashed marked object is the 'anchor' object of A.

Figure 6: The variation process.

Discussion

The first computational model about mental models has been presented in (Johnson-Laird & Byrne 1991). This model is able to parse relational premises and to insert objects into an array. This concept is related to our computational model, in fact, our model can be seen as a fundamental extension. The $\mathcal{CROS}$ has the following additional properties: a focus which performs the input operations (not outlined in (Johnson-Laird & Byrne 1991), the control function in which strategies for insertion principles can be defined, and one of the most important properties for qualifying and classifying the difficulty of tasks an implied complexity measure.

The models of (Schlieder & Berendt 1998) also make use of a focus and explain model preferences. Both models, however, are restricted to intervals as elements and a quite technical set of relations. A fundamental difference is that our model is much more natural because it uses solid objects and the most common verbal relations from natural language (and reasoning research). Our computational model shares the most features with the UNICORE model, which was developed in (Bara, Bucciarelli, & Lombardo 2001). Both models are based on the same three considerations: a model must include a grid of positions that are assigned to tokens (our spatial array), those tokens must have a name (our objects), and some objects may be in relation. The main difference between Baras model and the SRM model is that our model reproduces reasoning steps involved in spatial reasoning, whereas the UNICORE model does not have this property. Another advantage of the $\mathcal{CROS}$ model is that we have introduced a complexity measure, which explains the difficulty of reasoning problems.

Further research in spatial reasoning has been done by Barkowsky and colleagues with their diagrammatic processing architecture Casimir and MIRAGE (Bertel, Barkowsky, & Engel 2004; Barkowsky 2002). Their work focuses on the mental representation of spatial environments and interactions between external and mental diagrammatic representations. The model uses a representation of the working memory and activation. In this sense it is very similar to the $\mathcal{CROS}$. A difference is that the $\mathcal{CROS}$ focuses more on relational representations and is designed for deduction by means of mental models and for that the $\mathcal{CROS}$ explain complexity differences of different relational problems.

Starting from the question how to combine Baddeleys Working Memory Model and the preferred mental model theory based on recent cognitive results, we identified principles for a cognitive model consisting of two submemory systems, the PL and the VSSP, a semantic interpreter and introduced a control process based on a set of well-defined instructions able to manipulate the subsystems. This formalization of these processes is a first step in identifying the computational properties of the central executive. Take for instance a problem like (Ragni *et al.* 2006):

A is to the left of B.
C is to the right of A.
D is in front of C.
E is behind A.
Is A as near to D as C is near to E?

How is the question *A as near to D as C is near to E* be processed? Such a problem can be solved by a divide-and-conquer principle, that first the distance between A and D is determined and than the distance between C and E and then both distances have to be compared. But where in the WMM and how are the distances computed and compared? Since the distance information is a number, such an information is not stored in the VSSP. This distance information could then be stored in the phonological loop, but from the setting of the phonological loop, it is very unlikely that the comparison of distance information takes place in the phonological loop. There is a huge amount of psychological literature which makes it very precise that the phonological loop is a self-controlled process, where information can be stored but not be manipulated. For that reason it seems sensible to assume that there are (at least) two cells in Baddeleys WMM, which are used to apply such kind of operations. This existence is implicitly assumed in (Lemair, Abdi, & Fayol 1996). It is possible to speculate that these kind of cells might be used to perform not only operations like $<, =, >$ but also arithmetic operations like $+, -, *$. These kinds of implications of an algorithmic formalization of Baddeleys WMM are to be investigated next.

References

Baddeley, A. D. 1999. *Essentials of human memory*. East sussex, England: Psychology Press.

Baguley, T. S., and Payne, S. J. 1999. Memory for spatial descriptions: A test of the episodic construction trace hypothesis. *Memory and Cognition* 27.

Bara, B.; Bucciarelli, M.; and Lombardo, V. 2001. Model theory of deduction: a unified computational approach. *Cognitive Science* 25:839–901.

Barkowsky, T. 2002. *Mental Representation and Processing of Geographic Knowledge - A Computational Approach*, volume 2541 of *Lecture Notes in Computer Science*. Springer.

Bertel, S.; Barkowsky, T.; and Engel, D. 2004. The specification of the casimir architecture. Internal project report, R1-[ImageSpace], SFB/TR8 Spatial Cognition.

Byrne, R. M., and Johnson-Laird, P. N. 1989. Spatial reasoning. *Journal of Memory & Language* 28(5).

Goodwin, G. P., and Johnson-Laird, P. N. 2005. Reasoning about relations. *Psychological Review* 112.

Halford, G. S.; Wilson, W. H.; and Phillips, S. 1998. Processing capacity defined by relational complexity: implications for comparative, developmental, and cognitive psychology. *Behavioural Brain Science* 21.

Huttenlocher, J. 1968. Constructing spatial images: A strategy in reasoning. *Psychological Review* 75.

Johnson-Laird, P. N., and Byrne, R. M. J. 1991. *Deduction*. Hove (UK): Erlbaum.

Johnson-Laird, P. N. 2001. Mental models and deduction. *Trends in Cognitive Sciences* 5(10).

Knauff, M.; Rauh, R.; Schlieder, C.; and Strube, G. 1998. Continuity effect and figural bias in spatial relational inference. In *Proc. of the 20th CogSci Conference*. Mahwah, NJ: Lawrence Erlbaum Associates.

Knauff, M.; Mulack, T.; Kassubek, J.; Salih, H.; and Greenlee, M. W. 2002. Spatial imagery in deductive reasoning: a functional MRI study. *Cognitive Brain Research* 13(2):203–12.

Lemair, P.; Abdi, H.; and Fayol, M. 1996. The Role of Working Memory in Simple Cognitive Arithmetic. *European Journal of Cognitive Psychology* 8(1):73–103.

Manktelow, K. 1999. *Reasoning and Thinking*. Hove: Psychology Press.

Miller, G. A., and Johnson-Laird, P. N. 1976. *Language and Perception*. Cambridge: Cambridge University press.

Ragni, M.; Fangmeier, T.; Webber, L.; and Knauff, M. 2006. Complexity in spatial reasoning. In *Proceedings of the 28th Annual Cognitive Science Conference*. Mahwah, NJ: Lawrence Erlbaum Associates.

Ragni, M.; Knauff, M.; and Nebel, B. 2005. A Computational Model for Spatial Reasoning with Mental Models. In Bara, B.; Barsalou, L.; and Bucciarelli, M., eds., *Proc. of the 27th CogSci Conf.* Lawrence Erlbaum Associates.

Rauh, R.; Hagen, C.; Knauff, M.; T., K.; Schlieder, C.; and Strube, G. 2005. Preferred and Alternative Mental Models in Spatial Reasoning. *Spatial Cognition and Computation* 5.

Rips, L. 1994. *The Psychology of Proof*. Cambridge, MA: MIT Press.

Schlieder, C., and Berendt, B. 1998. Mental model construction in spatial reasoning: A comparison of two computational theories. In Schmid, U.; Krems, J. F.; and Wysotzki, F., eds., *Mind modelling: A cognitive science approach to reasoning*. Lengerich: Pabst Science Publishers. 133–162.

Vandierendonck, A.; Dierckx, V.; and Vooght, G. D. 2004. Mental model construction in linear reasoning: Evidence for the construction of initial annotated models. *The Quarterly Journal of Experimental Psychology* 57A:1369–1391.