Learning to reason about other people's minds

Lisette Mol

1220071

April - December 2004

Advisors:
dr. R. Verbrugge (Artificial Intelligence, RuG)
dr. P. Hendriks (Dutch, Artificial Intelligence, RuG)

Artificial Intelligence
Rijksuniversiteit Groningen

968.

Abstract

Two problems occur when trying to explain cognitive skill acquisition with the classical theory of skill acquisition [37]. One is that cognitive skills build on one another, which is not possible after skills are automated, because deliberate access to automated processes is limited. The second is that for some tasks, people are only able to describe why they are doing them in a certain way, after expert level performance has been reached [19]. This is not in accordance with the classical theory, in which experts make use of automated processes that are part of implicit memory.

Research on cognitive skill acquisition can lead to a better understanding of human cognition. More specific, it would be interesting to know what role reflective reasoning processes such as reasoning about others, metacognition and self-monitoring play in cognitive skill acquisition. This knowledge could be applied in the design of artificial tutors and conversational agents. As a first step in the right direction, this study has investigated to what extent people acquire and use complex skills and strategies in the domains of reasoning about others and natural language use, specifically, when playing the game Master(s)Mind(s).

An experiment was conducted in which participants played Master(s)Mind(s), a competitive head to head game (see chapter 4). In playing this game, it was beneficial to participants to have a mental model of the opponent, and to be aware of scalar implicatures. The complex skills that participants to this experiment could use were the application of their theory of mind, and reasoning from implicated meaning. A strategy to be developed was the strategy of being as uninformative as possible and to be aware that the opponent might do so as well. To achieve this, it was necessary for participants to be aware of the knowledge and desires of their opponent.

It was expected that participants would shift their language use from pragmatic to more logical, while repeatedly playing Master(s)Mind(s). Pragmatic language use can be described by Grice's quantity maxim, which is meant to be applied to cooperative conversation. It was expected that once people became familiar with the uncooperative context, people's language use would no longer be in accordance with this maxim. Therefore, pragmatic implicatures would no longer be used by participants, which would result in more logical language use.

Contrary to the predictions, most participants did not shift to a more logical language use. It was found that some participants made use of advanced cognitive skills like second order theory of mind use, logical language use and drawing pragmatic inferences, but participants did not seem to acquire these complex skills during the experiment. It can therefore be concluded that these skills can be transferred from other domains to the domain of playing Master(s)Mind(s), which suggests that they are not part of implicit memory.

No conclusive evidence was found for the hypothesis that playing Master(s)-Mind(s) and developing a strategy for it, can be seen as a form of dual-tasking. This also holds for the hypothesis that pragmatic language use results from an automated process, which can be overruled by a deliberate reasoning process, resulting in logical language use. Ideas to find conclusive evidence for these hypotheses are presented in section 6.3.

Contents

Chapter 1

Introduction

There are many situations in which people apply their knowledge about other people's minds, in a beneficial way. Examples can be found in playing games, teaching, negotiation, and communication. When playing poker for example, a player tries to find out how his opponents feel about their hands, to know whether he needs to raise a bet or to fold. In addition, the player tries to hide his own feelings, because he knows that his opponents want to know how he feels about his own hand, to be able to play more profitable themselves.

Here follows another example. When a general practitioner needs to explain to a patient that this patient suffers from a disease for which he needs to visit a specialist, he will do so very differently from when he explains to the specialist what is wrong with the patient. The general practitioner is able to keep in mind that the patient does not have as much medical knowledge as he himself does, whereas he can safely assume that the specialist does in this domain. Therefore, his explanation to the patient will contain fewer medical terms. In addition, the general practitioner will consider that the patient may react far more emotionally than the specialist does. This is because he assumes that the patient has the desire to be healthy, and to not have to worry about his health. He will therefore choose his words more carefully than he will when talking to the specialist.

In a more daily situation, when being asked for the location of a specific restaurant in his hometown, John will give a different explanation to a local than to a visitor. This is because he can realize that the visitor has very little knowledge about nearby streets and buildings, whereas the local has. He has to put less effort in explaining the restaurant's location to the local, because he does not need to imagine what it would be like if he would lack certain knowledge that he has. His knowledge in this domain, and the knowledge of another local in this domain are very similar as opposed to John's knowledge and the knowledge of the visitor.

The above examples illustrate that people are able to reason about the knowledge and desires of other people, and to apply this knowledge in daily life. In fact, reasoning about other people's beliefs, desires and intentions is often necessary for successful communication. Little is known however, about how people acquire skills that require this type of reasoning [37].

A well-known way to learn is to perform an action repeatedly, such that the effort one has to put into it decreases. Baking a cake could be learned this way, or adding two numbers. The second well-known way to learn is to generalize from examples. This way, children can learn to form regular past tenses. However, researchers from the ALICE research group at the University of Groningen assume that there is more to learning certain complex tasks than just these two types of learning. For example, reasoning about how one is performing while doing a certain task, or reasoning about other people's knowledge, may be necessary to learn certain tasks, or to achieve expert level performance in certain tasks.

The study described in this thesis is a pilot study on how people acquire skills that require reasoning about other people's minds, such as the tasks described in the above examples. More knowledge on this subject will be useful to better understand human cognition, and human skill acquisition. This understanding could lead to better ways of teaching. Not only human, but also artificial tutors need an image of what their students already master and where their difficulties lie. Artificial tutors can be based on human tutors, but to successfully do so it is necessary to understand how human tutors accomplish their success. In addition, knowledge on cognitive skill acquisition could help in writing instructions that facilitate learning.

Another application lies in the design of conversational agents. Since successful communication in natural language often depends on reasoning about other people's minds, conversational agents could benefit from being able to reason about people's minds in interacting with people. Knowledge on how humans learn and apply this type of reasoning can be very beneficial in the design of artificial agents.

In the study described, an experiment was conducted to investigate to what extent people use skills based on reasoning about other people's minds, when playing a game called Master(s)Mind(s). Three fields are brought together in this study: Cognitive Psychology, Pragmatics, and Logic.

Structure of this Thesis

In chapter 2, short introductions are given to the theories which are relevant for the present study. More detailed explanations can be found in the works referred to. Additionally, relevant work by scientists from the specific fields is described.

Chapter 3 describes the research question and the hypotheses formulated to investigate this research question. It is explained how the hypotheses link up with some of the theories described in chapter 2.

Chapter 4 is on the experiment that was performed to test the hypotheses. It describes the experimental method as well as the expected results. Some details about the experiment can be found in the appendices. Appendix A describes the implementation of the computer program that was used in the experiment.

Chapter 5 describes the results of the experiment, which are discussed in chapter 6. Chapter 6 also includes a comparison of the results found in the present work and the results found in earlier work described in chapter 2, and discusses what could be done in future work on cognitive skill acquisition.

In chapter 7, it is made explicit how modal epistemic logic and weak bidirectional optimality theory can be applied to the results found in the experiment. Chapter 8, which is the final chapter of this thesis, presents the conclusions of the study described.

Chapter 2

Theoretical Background

2.1 Theory of Mind

To have a theory of mind (ToM) is to understand that other people have desires, beliefs and thoughts much like one's own. People have mental models of others. These are models of the desires, beliefs and thoughts the other persons have. These models can be helpful in understanding and predicting others' behavior.

A mental model of another person can be described as being of a certain order. In a first-order model someone's beliefs, thoughts and desires are assumed to influence one's behavior. This, for example, enables us to distinguish between non-intentional actions of machines and intentional human actions.

In a second-order model it is also recognized that to predict others' behavior, the desires and beliefs that they have of one's self and the predictions of oneself by others must be taken into account. So, for example, you can realize that what someone expects you to do will affect his behavior. To have a third order model is to have a mental model of others holding a second-order model, etc.

Now, let us take a closer look at the different orders by which the knowledge which you have about another person can be described. Normal facts are of zeroth order. For example, 'his book is on the table' is zeroth order knowledge.

Once your knowledge is about the knowledge of other people, the order increases. 'He knows his book is on the table', and 'He does not know his book is on the table', for example, are first order knowledge. Negations do not increase the order of knowledge. 'He knows that I know his book is on the table' is second order knowledge, 'He knows that I know that he knows his book is on the table' is third order knowledge and so on.

In defining the different orders there are two points of interest. The first is that to increase the order, another agent must be involved. 'I know his book is on the table' and 'I know I know his book is on the table' are said to be of the same order. Another choice could have been made here, but for present purposes this leads to the most useful distinction. A motivation for this choice is that these statements are equivalent in the system S5 which is used in modal epistemic logic (see the following section). So for the order to increase, the agents the knowledge is about must be different.

An assumption made in S5 is that known facts are true. Thus, it follows from 'I know p' that p. This obviously does not hold the other way around, not everything that is true is known by me. Yet the choice is made to consider both 'I know p' and p to be zeroth order knowledge. This mainly is a matter of speech. The fact p in itself, which can be true or false, only becomes knowledge when it is known by someone. So only when someone knows that p, p can be considered zeroth order knowledge. Just as with 'he knows his book is on the table' the first 'I know' is left out. Only when I have the knowledge that he knows his book is on the table, the resulting 'I know he knows his book is on the table', can be considered first order knowledge.

From these two choices it follows that 'he knows I know he knows p' is third order knowledge whereas 'I know I know I know p' is zeroth order knowledge and 'he knows I know I know p' is second order knowledge just like 'he knows I know p'. In these examples p can be any zeroth order knowledge.

One's mental models of others can be useful in determining one's strategy in games. Imagine the following situation. John, Sadia, Chris and Mary are playing happy families. It is John's turn. John has the lion mother and he knows that none of the other players is aware that he owns a member of the lion family. John knows that Sadia owns two cards of the lion family. John also knows that Chris owns one of the lions, but he does not know which one. It is known to John that Sadia knows which lion Chris is holding.

Using his mental model of Sadia and Chris, John decides not to ask for a lion yet. This is because he knows that if he would make a wrong guess about who owns which member, Sadia could infer that he has the mother (since she knows which members she and Chris own) and could easily win the lions, which he knows she desires. So using his knowledge about the beliefs, desires and reasoning capacities of his opponents to predict their behavior, gives John a better chance of winning the game.

A theory of mind can also help to interpret language. Consider the following example. Mary has three sisters. When she encounters John, he tells her that he saw her sister today. Because Mary knows that John only knows one of her sisters, she can infer whom John is referring to. So she uses her model of John's knowledge to infer the meaning of his message.

Keysar, Lin and Barr [20] found that adults do not reliably use their theory of mind to interpret the actions of others. In one of their experiments one person followed the directions of another person to move objects around in a grid. One object was hidden in a bag by the person following the instructions and the director did not know the identity of this object. Still, when the description of the director more closely resembled the hidden object than one of the mutually visible objects, the follower often took it as the referent of the director's description and sometimes even moved the bag. So although the person following knew that the director did not know about the hidden object, he did not make use of this knowledge to interpret the directions. The knowledge that the director did not know about the hidden object is part of a first-order ToM. In other experiments by the same research group similar results were found [17], [21], [22].

2.2 Modal Epistemic Logic

Modal epistemic logic can be used to describe knowledge and beliefs of an agent, or a system of agents. In modal epistemic logic the K_i operator is used to represent that agent i knows something. For example K_1p, means *agent 1 knows p*. By definition an agent can only know things which are true. The K_i operator can take scope over an epistemic formula, for example $K_1(p \Rightarrow q)$ for *agent 1 knows that p implies q*, or K_1K_2p for *agent 1 knows that agent 2 knows that p*.

Especially the last example is of interest here. By nesting of the modal operator K_i, knowledge of different orders can be represented. This is relevant to describe knowledge of agents playing Mastermind. Mastermind is a two player game in which player 2 has to guess a secret code of four colors, that is composed by player 1. For each guess made by player 1, player 2 needs to specify how many colors from the guess match colors in the secret code, and how many of them are in the right place.

The fact that agent 1 has the first order knowledge that agent 2 knows that red occurs in agent 1's secret code of four colors could be represented by K_1K_2p, where p means *Red occurs in the secret code of agent 1*. Similarly, $K_1K_2K_1p$ would mean *agent 1 knows that agent 2 knows that agent 1 knows that red is in his secret code*. This is second order knowledge of agent 1. So the order corresponds to the number of K_i operators used, provided that the agent considered is the one named in the subscript of the first K_i operator and that that first K_i operator is left out (because it only specifies which agent has the knowledge and is not part of the knowledge itself). Additionally, each K_i operator has to have a different agent as a subscript (this corresponds to the requirement of agents being different described in the previous section).

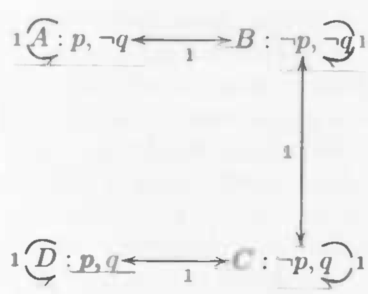
A useful semantics for modal epistemic logic was formalized by Kripke [24] and is called *possible world semantics* or *Kripke semantics*. In this semantics an agent's view of the world is expressed by the worlds he thinks to be possible according to the available information. For example, if agent 1 knows that agent 2's secret code contains either red or yellow, he only thinks worlds in which agent 2's secret code contains red and worlds in which agent 2's secret code contains yellow to be possible. If in addition, he knows that agent 2's secret code contains blue, this has to be the case in all worlds considered possible by agent 1. For if there would be a possible world in which agent 2's code did not contain blue, agent 1 could not be sure it did. Thus, for an agent to know something, means that it is true in all worlds the agent considers possible.

In addition to the K_i operator, the M operator can be used to represent what an agent thinks that might be. M_1p holds if p holds in at least one world which agent 1 thinks to be possible. In a two-valued, classical logic M_1p is equivalent to $\neg K_1 \neg p$.

When looking at a finite system of multiple agents, there are two more useful operators. E, for *everyone knows that* and C, for *it is common knowledge that*. Agents are said to have common knowledge of p if it is the case that everyone knows that p, everyone knows that everyone knows that p, everyone knows that everyone knows that everyone knows that p, etc. ad infinitum.

What an agent thinks to be possible can depend on the current state of the world. Agent 1 might know for example, that if agent 2's code contains red, it cannot contain blue. Let p denote *agent 2's code contains red*, and let q denote *agent 2's code contains blue*. Now, the knowledge of agent 1 described above can be represented by $K_1(p \rightarrow \neg q)$. This means that from the current state of the world, agent 1 would only consider worlds where $p \rightarrow \neg q$ holds to be possible. Example 1 shows that in different worlds, $K_1(p \rightarrow \neg q)$ can have a different truth value.

Example 1



p: Agent 2's code contains red.
q: Agent 2's code contains blue.

Figure 2.1: example 1

In figure 2.1, there are four worlds: A, B, C, and D. In world A p and $\neg q$ hold, in world B $\neg p$ and $\neg q$ hold , etc. The arrows from a world represent which worlds the agent would consider possible if that original world were the current state of the world. If world A would be the current state of the world, agent 1 would consider world A and world B to be possible, since there are arrows from world A to world A, and from world A to world B.

Suppose the current world is A. In world A, it is true that agent 1 knows that if agent 2's secret code contains red, it does not contain blue. More formal, in world A, $K_1(p \rightarrow \neg q)$ holds. For an agent to know something, it must be true in all worlds the agent considers possible, in other words, in all worlds that are accessible to that agent. From world A, world A and B are accessible to agent 1. $p \rightarrow \neg q$ is logically equivalent to $\neg p \vee \neg q$. In world A and in world B $\neg q$ holds, and thus $p \rightarrow \neg q$ holds in all worlds accessible to agent 1 from world A. It can therefore be concluded that $K_1(p \rightarrow \neg q)$ holds in world A.

Now suppose that the current world is B. In world B, $K_1(p \rightarrow \neg q)$ is true as well. In addition to world A and B, where $p \rightarrow \neg q$ holds, as was already shown, world C is accessible to agent 1 from world B. In world C $\neg p$ holds and thus $p \rightarrow \neg q$ holds in world C. Hence, in all worlds accessible to agent 1 from world B, $p \rightarrow \neg q$ holds and thus $K_1(p \rightarrow \neg q)$ holds in world B.

In world C however, $K_1(p \rightarrow \neg q)$ does not hold. This means that if world C is the current world, agent 1 does not know that if agent 2's secret code contains red, it does not contain blue. This is because from world C, world D is accessible to agent 1. In world D neither $\neg p$ nor $\neg q$ holds, and thus $p \rightarrow \neg q$ does not hold in world D. Therefore $p \rightarrow \neg q$ does not hold in all worlds accessible to agent 1 from world C, and thus $K_1(p \rightarrow \neg q)$ does not hold in world C. In analogy, $K_1(p \rightarrow \neg q)$ is not true in world D.

Example 2

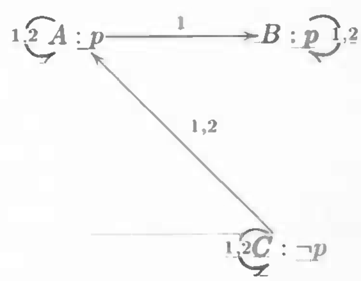


Figure 2.2: example 2

In figure 2.2 there are three worlds and accessibility relations are specified for two different agents. The labels above the arrows indicate for which agent the accessibility relation holds. Suppose the current world is A. If in all worlds accessible to agent 1 K_2p holds, agent 1 knows that agent 2 knows p. This is true in world A of figure 2.2. From world A, world A and B are accessible to agent 1. From world A, only world A is accessible to agent 2 and p holds in world A, thus K_2p is true in world A. In world B, only world B is accessible to agent 2 and in B p holds, thus K_2p holds in world B. Thus, K_2p is true in each world that is accessible to agent 1 from world A and therefore K_1K_2p holds in world A.

In a multiagent system each agent has its own knowledge. So each agent has some worlds which it considers possible and each agent has its own accessibility relations, which specify which worlds the agent thinks possible given a certain world.

Within modal epistemic logic several axiomatic systems can be used. One of them is the system S5 in which the following axioms hold:

(A1) All (instances of) propositional tautologies.

(A2) $(K_i\phi \wedge K_i(\phi \rightarrow \psi)) \rightarrow K_i\psi$ (i=1, ..., m)

(A3) $K_i\phi \rightarrow \phi$ (i = 1, ..., m)

(A4) $K_i\phi \rightarrow K_iK_i\phi$ (i = 1, ..., m)

(A5) $\neg K_i\phi \rightarrow K_i\neg K_i\phi$ (i = 1, ..., m)

In addition, the following derivation rules can be used:

(R1) $$\frac{\phi \quad \phi \rightarrow \psi}{\psi}$$

(R2) $$\frac{\phi}{K_i\phi}$$ (i=1, ..., m)

(A4) and (A5) state that an agent knows that he knows something and that an agent knows that he does not know something. In reality, these axioms do not always hold for human beings. From (A3) and (A4) it follows that the implication in (A4) holds in both directions. From (A3) and (A5) it follows that the implication in (A5) holds in both directions as well.

A more formal and complete explanation of modal epistemic logic can be found in [35].

2.3 Human Reasoning

In addition to reasoning about others, as described in the previous sections, people have the ability to reason about their own intentions, beliefs, desires and actions. This type of reasoning is called reflective reasoning. An example of reflective reasoning is self-monitoring, which means critically watching one's own performance while doing a certain task.

Human reasoning has often been divided into automatic and controlled processing. Automatic reasoning is unconscious and is usually modeled in a connectionist architecture. Controlled reasoning is conscious and is usually modeled in symbolic systems.

In [31], Schneider and Chein write that Schneider and Shiffrin [32] defined automatic processing as the activation of a sequence of nodes that "nearly always becomes active in response to a particular input configuration", and that "is activated automatically without the necessity for active control or attention by the subject". Automatic processes "require an appreciable amount of consistent training to develop fully" (p. 2).

Controlled processes on the other hand are defined by Schneider and Shiffrin [32] as "a temporary sequence of nodes activated under control of, and through attention by, the subject." They are "tightly capacity limited, but the costs of this capacity limitation are balanced by the benefits deriving from the ease with which such processes may be set up, altered, and applied in novel situations for which automatic sequences have never been learned." (p. 2-3)

The classical theory of skill acquisition describes learning as a process of automation: one starts a new skill in the cognitive stage (stage 1), in which controlled, deliberate reasoning is needed to perform the task. This stage is characterized by slow performance and errors. By repeatedly performing the skill, eventually the autonomous stage (stage 2) is reached, where performance is fast and automatic, requiring little working memory capacity.

In [34], Sun and Zhang stress the interaction of implicit (automatic) and explicit (controlled) reasoning processes during skill acquisition. In [10], Evans describes that most reasoning tasks have automatic and deliberate components. The two types of reasoning thus seem to be closely intertwined.

Verbrugge, Hendriks, Taatgen et al. [37] consider skill acquisition as a continuous interplay between deliberate and automatic reasoning processes. In their view, the classical theory of skill acquisition has two important limitations:

1. Skills are usually considered in isolation, whereas in reality they build on one another.

   For example, the skill of multiplication is based on the skill of addition. However mastered and hence automated skills cannot in themselves serve as a basis for more advanced skills, because deliberate access to automated skills is limited. Hence, it remains unclear how transfer of knowledge from one skill to another is possible.

2. The capacity for deliberate reasoning sometimes increases rather than decreases when becoming an expert.

   Karmiloff-Smith [19], for example, reports that children can only describe what they are doing after they have mastered a skill (e.g., in number conservation experiments). This cannot be explained by assuming skill acquisition to end at stage 2.

They aim to address these shortcomings by considering skill acquisition as a continuous interplay between deliberate and automatic reasoning processes, ultimately leading to a third stage of skill. They assume that to reach this third stage (expert level performance), reflective deliberate reasoning processes, such as self-monitoring, are crucial in many domains of cognition.

This study aims to investigate whether people can learn to use such reflective deliberate reasoning processes, when playing the game Master(s)Mind(s) (see 4.1).

2.4 Scalar Implicatures

This section first provides a short introduction on meaning in natural language. Two perspectives are mentioned: meaning analyses based on truth-conditional semantics and pragmatic meaning. Then a special case of pragmatic meaning is introduced: the scalar implicature. This is followed by a description of Grice's explanation for this type of implicature, in terms of his quantity maxim. The section ends with the presentation of several experimental results on scalar implicatures.

Meaning

When determining the meaning of an utterance, the conditions in which the utterance is true can be considered. For example, if someone says:

1. It is raining,

this is true if and only if it is the case that it is raining. In all other cases 1 is false. So the meaning of the utterance could be defined as what the world would have to be like for it to be true. This is an analysis of meaning according to truth-conditional semantics.

Now consider the following dialog.

2. Shall we go for a walk?

1. It is raining.

The meaning of utterance 1 in this dialog is not only that it is raining, it also implies that the speaker does not want to go for a walk. This implicated meaning can not be determined when looking at sentence 1 in isolation. It is not determined by the words or the grammar of a language, but by the context in which the utterance is spoken and conversational conventions. This type of meaning is called pragmatic meaning.

Scalar implicatures

In natural language, the interpretation of a sentence containing a scalar term often goes beyond its literal meaning. A scalar term is an item from a set ordered in informational strength, for example ⟨some, most, all⟩. This ordered set of alternatives is called a scale. Here follows an example.

3. Some students passed the test.

4. Not all students passed the test.

Sentence 3 is logically true if and only if at least one student passed the test. So according to truth-conditional semantics it would be true in a world where all students passed the test. However, when you are told 3, it's quite natural to infer from 3 that 4 holds. So the pragmatic meaning of 3 differs from its truth-conditional meaning.

This example is an example of a scalar implicature. In case of a scalar implicature it is communicated by a weaker claim (using a scalar term) that a stronger claim (using a more informative term from the same scale) does not hold. A stronger term excludes more possibilities than a weaker term, therefore it is said to be more informative.

Grice [13] gave an explanation of conversational implicatures in terms of maxims. He suggested that under normal circumstances communication is governed by these maxims, which state that interlocutors are expected to offer contributions which are truthful, informative, relevant to the goals of the conversation and appropriately phrased. So according to Grice, communication is essentially cooperative. One of his maxims is the quantity maxim:

i. Make your contribution as informative as is required.

ii. Do not make your contribution more informative than is required.

Grice suggested that people use this maxim to infer the implicit meaning of a sentence. So if someone tells you that some students passed the test, you can infer that not all students did so, because if this would have been the case the speaker probably would have known. Assuming that he acts according to the quantity maxim and thus is as informative as possible, he would then have used a stronger term. In interpreting scalar implicatures the hearer thus considers a set of ordered alternatives (a scale) that the speaker could have used. Some examples of scales are: ⟨some, all⟩, ⟨if, iff⟩, ⟨possibly, necessarily⟩, ⟨believe, know⟩, ⟨or, and⟩, ⟨start, finish⟩. These scales are ordered from weak to strong.

Experimental Results

In an experiment by Papafragou and Musolino [30] subjects were presented with contexts which satisfied the semantic content of the more informative terms but were described using the less informative terms. They found that adults overwhelmingly rejected these infelicitous descriptions whereas children in the age of 5 to 6 almost never did so. By manipulating the contexts such that they more readily invited scalar inferences, children's performance improved, but still remained worse than adult performance.

Children differ in the production and interpretation of scalar terms. Although they use the appropriate terms in production, their interpretation often is more purely logical compared to adults, thus missing scalar implicatures [29].

Feeney et al. [11] conducted an experiment in which undergraduate students performed a computerized sentence verification task. They recorded the student's answers and reaction times. Here are two of the *some* sentences they used.

1. Some fish can swim.

2. Some cars are red.

They found that reaction times for logical responses to infelicitous *some* sentences such as 1, were longer than those for logically consistent responses to felicitous *some* sentences as 2. Notice that to both sentences the logical response is 'true'. The pragmatic response to 2 is 'true' as well. The pragmatic response to 1 is 'false'. So the sentences in which the logical and pragmatic response are in conflict result in longer reaction times.

In the experiment three students gave pragmatic responses only, 21 students gave a mixture of pragmatic and logical responses and 25 gave logical responses only. In the group of students that gave both logical and pragmatic responses, no type of response was found to take significantly longer than the other. Thus, the group that gave logical responses only mainly caused the longer reaction times for logical responses to infelicitous *some* sentences.

These results favor a theory that logical responses are due to inhibition of a response based on the pragmatic interpretation over a theory that logical responses result from failure to make the pragmatic inference.

2.5 Bidirectional Optimality Theory

Optimality Theory

Optimality Theory (OT) can be used to model a speaker's linguistic knowledge. It has successfully been used in phonology [26], morphology [18], syntax [28], and semantics [15]. In OT an optimal solution is selected from a set of candidates. There is a mechanism called *Gen* (for generator) which generates candidate outputs for a given input. These candidates are evaluated by a mechanism called *Eval* (for evaluator). Eventually one of the candidates is ranked optimal.

Eval consists of a set of ordered constraints. These constraints can be compared to rules in other linguistic theories. An important difference is however, that optimal candidates are allowed to violate constraints. A weaker constraint can be violated in order to satisfy a stronger constraint. Thus constraints are ordered hierarchically from strong to weak and represent general tendencies in language rather than strict rules. The candidate which violates the fewest, weakest constraints is the optimal candidate. Here follows an example.

Suppose there are four constraints: 1, 2, 3, and 4 of which 1 is the strongest, followed by 2, etc. Candidate A violates constraint 1, candidate B violates 2, and candidate C violates 3 and 4. This situation is shown in table 2.1. In this case, candidate C would be the optimal candidate, because the constraints violated by the other candidates are stronger than the constraints violated by candidate C. So even though candidate C violates more constraints than the other candidates, it is ranked optimal. Would there be a candidate D, which only violates constraint 3, than this candidate would be favored over candidate C, because it violates fewer constraints and no constraints that are stronger than the constraints violated by candidate C.

Table 2.1: Overview of constraints violated by candidates.

	1	2	3	4
Candidate A	*			
Candidate B		*		
Candidate C			*	*

It is assumed that there is one set of constraints which can be used to describe all human languages. The differences between languages result from a different ordering of the constraints. All linguistic phenomena, whether in the field of phonetics, phonology, morphology, etc. are believed to be governed by one ordered set of constraints. In practice, when studying a certain aspect of language, only a few constraints and their relative ordering are described by a theory. The ordering of these constraints as to the other constraints is thought to be irrelevant to the studied phenomenon.

A constraint can either prohibit or demand something [25]. For example, a constraint could demand that all speech segments in the output must have a correspondence in the input (FaithV), or that a clause has a subject in canonical position [14]. Constraints should not compare different candidates themselves, because this is part of Eval.

Bidirectional OT

In sentence interpretation two perspectives can be taken: the view of the speaker (expressive perspective) and the view of the hearer (interpretive perspective) [15]. In bidirectional optimality theory both of these perspectives are taken into account.

It is assumed that both the speaker and the hearer want to minimize their effort in conversation. This is stated by two principles, which are a reduction of Grice's maxims and were first proposed by Atlas and Levinson [2]. The Q-principle is about minimizing auditor's effort and corresponds to the first part of Grice's quantity maxim (see section 2.4), while the I-principle seeks to minimize speaker's effort and corresponds to the second part of the quantity maxim, as well as to several other maxims [6] and [7].

According to Blutner and Solstad ([6], [7]), the I-principle compares different possible interpretations for the same syntactic expression and the Q-principle compares different possible syntactic expressions that the speaker could have used to communicate the same meaning.

Thus, an optimization takes place over pairs of form (f) and meaning (m). Without bidirectionality, optimization would be over either form or meaning, but not both. Blutner and Solstad propose two versions of bidirectional optimality theory: a strong and a weak version. Pairs are related by means of an ordering relation $>$, which means *being more harmonic*. This relation is determined by the constraints which are proposed by a theory.

Strong Version

A form-meaning pair (f, m) is optimal iff it is realized by Gen and it satisfies both the Q- and the I-principle, where:

(Q) (f, m) satisfies the Q-principle iff there is no other pair (f', m) realized by Gen such that $(f', m) > (f, m)$

(I) (f, m) satisfies the I-principle iff there is no other pair (f, m') realized by Gen such that $(f, m') > (f, m)$

Weak Version

A form-meaning pair (f, m) is super-optimal iff it is realized by Gen and it satisfies both the Q- and the I-principle, where:

(Q) (f, m) satisfies the Q-principle iff there is no other pair (f', m) realized by Gen which satisfies the I-principle such that $(f', m) > (f, m)$

(I) (f, m) satisfies the I-principle iff there is no other pair (f, m') realized by Gen which satisfies the Q-principle such that $(f, m') > (f, m)$

In the strong version the optimal pair consists of the optimal form and the optimal meaning. Weak optimization takes place in rounds. The first round determines the optimal candidate, the following rounds each determine a super-optimal candidate. The first round of weak optimization is strong optimization. Each round all candidates having the same form, but a less optimal meaning, or the same meaning, but a less optimal form, as the optimal candidate of the preceding round are excluded and optimization takes place over the remaining candidates. This way, several superoptimal pairs are allowed.

Theoretically, there is an infinite number of superoptimal pairs, each with a different form and meaning. Linguistically, this yields incorrect predictions [3]. Psychologically, this is not plausible. Since the capacity of the human brain is limited, the number of rounds should somehow be restricted. This is analogous to recursion in sentence structure. For example, linguistically, an infinite number of embedded phrases is allowed. This number is restricted however, by the limited processing capacity of the human brain.

The weak version can explain marked expressions having a marked interpretation, although both the expression and the situations they describe have a more efficient counterpart [6]. Marked expressions are expressions that are special, for example because they occur far less frequent than expressions that are not marked. Consider the following example.

John sees *himself/ him.*

The bidirectional analyses of this example is similar to the example Hendriks and Spenader give in tableau 3 of [16]. *Himself* could be used to refer to *John*, or to someone not mentioned in this sentence. Instead of *himself*, *him* could be used and this could also refer to *John* or to someone else. These possibilities are listed as candidates in table 2.2. ⟨himself, coref⟩ means that *himself* is used to refer to John, ⟨himself, disjoint⟩ means that *himself* is used to refer to someone else, etc. The candidates are the form-meaning pairs over which optimization takes place.

Table 2.2: OT Tableau for 'John sees *himself* / *him*.

	candidates	C1	C2
I	1. ⟨himself, coref⟩		
	2. ⟨himself, disjoint⟩	*	
	3. ⟨him, coref⟩		*
II	4. ⟨him, disjoint⟩		*

The first column indicates in which round a candidate is optimal. The last two columns indicate which constraints are violated by the candidates. An asterisk marks a violation. C1 and C2 are constraints of which C1 is ranked higher. The meaning of the constraints is not relevant for the present example, but they could be the constraints used in [16]: Principle A (C1), and Referential Economy (C2).

Since candidate 1 violates the fewest, weakest constraints (in this case no constraint), candidate 1 is the optimal candidate. This is the only possibility allowed by strong bidirectional OT. Using weak bidirectional OT, we continue with a second round. Candidate 2 and 3 are eliminated, since 2 has the same form and 3 has the same meaning as candidate 1 and candidate 1 is the optimal candidate of the first round. This leaves candidate 4 as the optimal candidate of the second round. Thus, the weak version allows the use of *himself* to refer to *John* and the use of *him* to refer to someone else. The strong version cannot explain the use of *him* in this sentence whereas the weak version can.

Bidirectional OT and ToM

The speaker and the hearer both know the constraints and their ordering, because they are speakers of the same language. In addition, it is common knowledge that every speaker of this language uses this particular ordered set of constraints. This common knowledge is the same for all communication. This makes it unlikely that this common knowledge is inferred over and over again, each time a hearer wants to interpret a certain term. It could simply be remembered. This is not to say that the optimization process itself does not have to take place for each instance.

According to bidirectional OT, to interpret utterances the hearer must have a mental model of speakers in general. The hearer needs this to determine which form-meaning pair is optimal for the speaker, given the form of the utterance and the constraints the speaker uses.

For example, if the speaker uses *him* the hearer can infer that he does not mean *John*. This is because the hearer knows that if John was meant, the speaker would have used *himself* since it is optimal. Thus, given the common knowledge that every speaker uses the same ordered set of constraints, the hearer can use first order ToM reasoning to understand the utterance.

Though it might not be obvious, common knowledge of the constraints is needed in communication. Imagine the hearer not knowing that the speaker knows that the hearer knows the constraints. So the hearer lacks second order knowledge. If the speaker does not know that the hearer knows the constraints, he could not be sure the hearer would understand him when he would use them.

So in this case there would be no reason for the hearer to assume the speaker has used the constraints in formulating his utterance. Thus, the hearer cannot use the constraints in interpreting the utterance and communication will not be successful. Similar arguments can be given for higher order reasoning.

Many scientists, also outside the field of OT, believe that common knowledge is necessary for successful communication, for example Clark and Marshall [8]. The intuition behind this idea is that when you use a reference you should be sure that you and the addressee know what it refers to, know that you both know what it refers to, know that you both know that you both know what it refers to and so on ad infinitum.

2.6 Game Theory

According to game theory, a game is being played whenever people interact with each other. Game theory provides a way to formally describe and categorize games and strategies. It has successfully been applied to problems in economics, political science, biology and social philosophy.

A strategy in game theory is defined in a formal way. To understand strategies, let us first look at games. A game must have rules, which specify for each player what can be done at what point in time. In addition, the rules should specify how the players are rewarded at the end of the game.

A game can be represented as a game tree. The root of the tree represents the first move of the game. The leaves correspond to possible game endings and should be labeled with the outcomes of the game for each player. Each node represents a possible move and the edges leading away from a node represent the available choices, or the actions possible at that point in the game. For each node, it should be specified which player can make a choice or perform an action. If the move depends on chance, for example when rolling dice, the corresponding node is assigned to Chance (or Nature) and the edges leading away from that node must be labeled with a probability. Such moves are called chance moves. A play then consists of a connected chain of edges, from the root to one of the leaves.

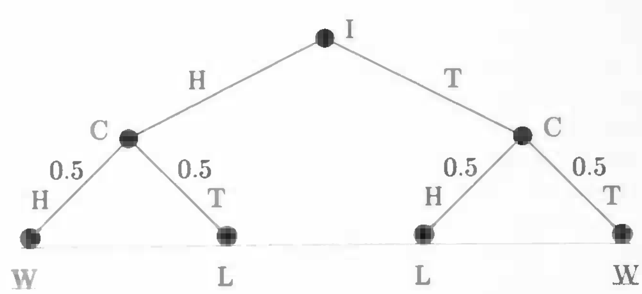


Figure 2.3: Game tree for the game Heads or Tails.

Figure 2.3 is an example of a game tree for the simple game Heads or Tails. First, player I needs to choose either Heads or Tails. Then, there is a chance move, a coin is flipped so there are two edges with probability 0.5 leading away from the chance nodes. These each end in two leaves, one in which the player wins, and one in which he loses.

A *pure* strategy for player i can now be defined as 'a statement that specifies an action at *each* of the decision nodes at which it would be player i's duty to make a decision if that node were actually reached', [17] p30. Thus, in the Heads or Tails example a strategy for player one would need to specify whether player one would choose Heads or Tails. A possible strategy is to choose Heads.

The knowledge a player has at a particular time that is relevant to the game is represented in game theory as an *information set*. Information can either be *perfect* or *imperfect*. In bridge, information is imperfect, because the players do not know the hands of the other players. This is different from the game of chess, where each player knows everything there is to know about the current state of the game, at each point in the game, because both players can see the board and the positions of the pieces. Chess is a game of perfect information. Under perfect information, each player knows exactly where he is in the game tree at each point in the game, and no moves are simultaneous. In a game of imperfect information, a pure strategy needs to specify what decision a player takes, or what action a player performs at each possible information set.

The above Heads or Tails example considers a very simple game, with very simple strategies. The game used in the experiment described in this thesis (see section 4.1), a version of Master(s)Mind(s), is far more complex. Master(s)Mind(s) is a two-player *zero-sum* game. *Zero-sum* means that the pay-offs of all players always sum to zero. In Master(s)Mind(s) this is the case, because if one player wins, the other player loses. A two-player zero-sum game is necessarily strictly competitive (see [17] p237, 238). In the game used, each player has a secret code of four different, ordered colors, and has to guess the secret code of the opponent in order to win. This can be done by making guesses and receiving feedback sentences on how many colors are right, and how many of them are in the right position.

The game tree for the version of Master(s)Mind(s) used would become extremely large, because there are many possible ways for the game to evolve. In addition, the information players have depends on what choices their opponent makes, which results in many possible information sets. It would thus be very impractical to define a formal strategy for this version of Master(s)Mind(s). However, global strategies can be described for this game such as 'revealing little information' or 'concentrating on guessing the opponent's secret code'. In the following chapters, *strategy* is used in this less formal way.

In the game Master(s)Mind(s), information is not *symmetric*, which means that a player has information different from the other players when he moves or at a leaf. In fact, both players have information that is different from the information that the other player has, since both players have their own secret code, which they know, but their opponent does not know. Because the information is not symmetric, it is also imperfect. In Master(s)Mind(s) there are also simultaneous moves, because both players have to choose feedback sentences at the same time.

In game theory, players are assumed to be rational and capable of perfect reasoning. For Master(s)Mind(s), this is not a plausible assumption. It would imply that all players would be capable of infinitely high order Theory of Mind reasoning. Game theory has often been criticized for the assumption of perfect reasoning [12] p21. Apart from infinitely high order ToM reasoning, and more general infinite reasoning capacities, it would also predict that agents can acquire an infinite amount of information and can even predict the future. In [33], Simon has introduced the term *bounded rationality* to describe the limited cognitive resources and capabilities of the human mind. In [12], Gazendam explains different ways of interpreting bounded rationality.

Because Master(s)Mind(s) is a game of imperfect information, the players cannot know what the opponent knows. This makes it more difficult to carry out first order ToM reasoning. Instead of a reasoning pattern like 'he knows A, thus he will act B', players have to use patterns like 'he acts B, if he would know A he may act B, thus he may know A' to infer facts about the opponent's knowledge, which contains the secret code.

Game theory has been applied to bidirectional OT by Dekker and Van Rooy [9]. They found that an optimal form-meaning pair in weak bidirectional OT can be reformulated as a Nash equilibrium in OT. A Nash equilibrium is a very well known solution concept in game theory. It can be defined as follows: "If there is a set of strategies with the property that no player can benefit by changing her strategy while the other players keep their strategies unchanged, then that set of strategies and the corresponding payoffs constitute the Nash Equilibrium." From http://william-king.www.drexel.edu/top/eco/game/nash.html

Chapter 3

Research Question and Hypotheses

This study originally served as a pilot for a larger project at the University of Groningen. In this larger project, it is assumed that cognitive skill acquisition can be explained through the same cognitive mechanisms, within three different cognitive domains: reasoning about others, pragmatics and instruction. The problem statement in this larger project is: 'How do deliberate and automatic processes interact in the acquisition of complex skills?' In this study, the focus is not so much on how humans acquire and use complex skills, but on to what extent they do so.

In this study, two domains are addressed: reasoning about others and pragmatics. As described in section 2.1, a dissociation has been found in the reflective ability to distinguish one's own beliefs from those of others, and the application of this knowledge. Yet the application of first order theory of mind would often lead to better strategies and better performance. For example, in the experiment of Keysar at al. [20] people would have to consider fewer objects if they would realize that the director could not see and thus could not know about certain objects, and this could lead to faster performance. In the domain of pragmatics, the application of ToM reasoning could have an influence on interpretation and production.

A question still to be answered is whether reflective cognition and the application of the knowledge in which it results, are necessary for cognitive skill acquisition. As a step in the right direction, the following research question is stated for this study.

Research Question *To what extent do people acquire and use complex skills and strategies in the domains of reasoning about others and language use, specifically, when playing the game Master(s)Mind(s)?*

The assumption is made that to acquire and use complex skills, reflective reasoning is indeed necessary. To investigate how deliberate and automatic processes interact in the acquisition of complex skills, the following hypothesis is adopted:

Hypothesis 1 *Performing a task and simultaneously reflecting upon this task can be seen as a form of dual tasking.*

This hypothesis has been adopted from Verbrugge, Hendriks, Taatgen et al. [37]. In their view, reflective deliberate reasoning processes, such as self-monitoring, are crucial for reaching expert level performance in many domains of cognition, including learning from instruction, pragmatics and reasoning about others.

The hypothesis is that when people perform a task which involves reasoning with incomplete information or drawing pragmatic inferences, reflection can be considered as a second task. The first task includes reasoning based on one's own knowledge and the truth-conditional (e.g., logical) meaning of utterances. The second task is more complex and includes using reflection to reason about others and to infer from implicated meaning.

When playing Master(s)Mind(s), a two player version of Mastermind (see 4.1), the first task is to play the game according to its rules. This involves reasoning about the game rules and determining which sentences are true. The second task is to develop a winning strategy for the game. This involves reasoning about what your opponent thinks, is trying to make you think or thinks that you are trying to make him think as well as determining what is implicated by an utterance or which utterances reveal the least information while still being true.

Two additional hypotheses have been formulated, which are relevant to the domain of language use and also involve reasoning about others. To explain hypothesis two, an explanation of cooperative and uncooperative conversation is given first.

Grice's quantity maxim states that people try to be as informative as possible, yet not more informative than is required. In cooperative conversation this is a reasonable assumption, but sometimes conversation is not fully cooperative. For example people can hold back information, deceive or even lie. In these situations people obviously are not giving as much information as possible.

Instead of always being easily mislead and deceived, people can anticipate for being wrongly informed, especially when the context provides them with reasons to be suspicious. Consider the following two situations.

Situation 1 You are called by a friend who asks you for a phone number. You know the number by heart, so you ask her whether she has pen and paper. She answers you with 'No, I don't'. Can you conclude that she also does not have a pencil and paper ready?

Situation 2 You are playing happy families and you are the first to pose a question. You ask your opponent for the mother of a certain family. Your opponent replies with 'not at home'. Can you conclude that he doesn't have any members of this family?

Situation 1 is an example of cooperative communication as opposed to situation 2. If we apply Grice's quantity maxim to situation 1, it can indeed be concluded that your friend also does not have a pencil. For if she did, she would have told you so since it is easy for her to infer that it would be relevant.

In situation 2 your opponent can easily infer that you would be interested to know whether he has any other cards of the same family. Yet he does not tell you so, because he is playing a competitive game and is thus not being cooperative. However, you are perfectly aware of this so you do not draw the pragmatic inference that you would draw in a cooperative situation.

Hypothesis 2 states that this also holds for using scalar terms like *some* and *most*. It is expected that once people are familiar with the uncooperative situation, they will make less strong pragmatic inferences than they do in cooperative situations. This will not only influence their interpretation, but also their production of sentences with scalar terms.

Hypothesis 2 *In an uncooperative conversation, people will shift their interpretation and production of quantifiers from pragmatic (using Grice's quantity maxim) to less pragmatic (not using Grice's quantity maxim).*

Hypothesis 3 is on what kind of reasoning is involved in using quantifiers, especially to make the shift described in hypothesis two. In [11], Feeney et al. propose that there are three stages to people's interpretation of *some*. The first is the logical interpretation which precedes children's sensitivity to scalar implicatures. The second stage is the pragmatic interpretation which results from drawing pragmatic inferences. This is in line with the results found by Noveck [29] and Papafragou & Musolino [30]. Feeney et al. found evidence for a third stage, in which adults can choose a logical interpretation over a pragmatic interpretation, even though they can make the pragmatic inference that *some* implies *not all* (see section 2.4).

The theory of three stages that Feeney et al. propose seems in line with the three stage model proposed by Verbrugge, Hendriks, Taatgen et al.(see section 2.3). If so, the process of making pragmatic inferences should be an automatic process and the ability to overrule this pragmatic interpretation would probably be a deliberate reasoning process in which one's theory of mind is used. To investigate this, hypotheses 3 is formulated.

Hypothesis 3 *In interpreting and producing quantifiers, people make use of an automated process, which results in a pragmatic use of the quantifier. This automatic process can be 'overruled' by a deliberate reasoning process, which results in a logical use of the quantifier.*

Chapter 4

Experiment

To test the hypotheses, an experiment was conducted in which human participants played a head to head game: Master(s)Mind(s). To facilitate data collection, a Kylix implementation of the game Master(s)Mind(s) was made (see Appendix A), and participants played the game on two connected computers. Kylix is an environment for object oriented programming, which can be used on a Linux platform. It is similar to Delphi, which can only be used on a Windows platform. While playing the game, all the choices participants made were recorded in a file, as well as their answers to questions that they were asked during the game through pop-up screens. Analyzing these files led to the results presented in chapter 5.

This chapter starts with a description of the game that participants played, and an explanation why the game was designed in this way. Next, in the section Design, there is an explanation of how different ways of language use lead to different behavior in this game. This explanation is followed by the predictions. Finally, the procedure, used materials, and participants are described.

4.1 Experimental Setup

Game

Subjects will play a competitive game in pairs, which is based on a two player symmetric version of the game Mastermind, called Master(s)mind(s) [23]. In this game each player can choose a secret code. The goal of the game is to be the first to guess the other player's secret code. The subjects will both use a computer to play the game; these computers will be linked to enable the players to compete against each other.

Each turn, one player (player 1) can make a guess about the code of the other player (player 2). This guess has to be a possible secret code. Player 2 has to evaluate this guess by submitting how many colors are correct and in the right place and how many colors are correct, but in the wrong place, compared to his secret code. This evaluation will not be visible to player 1 and will be checked by an algorithm, which will correct the participant if necessary.

Player 2 then has to provide feedback to player 1 by communicating two sentences, which can be chosen from a list of sentences differing in pragmatic strength (see Appendix B). The sentences are in Dutch. One sentence must be about the colors guessed and one about the positions of the colors. Both sentences must be true.

In response to this, player 1 will submit her interpretation of the feedback in a code of four items. A black dot will be used to represent a right color in the right place, a white dot to represent a right color in a wrong place and an empty item to represent a wrong color. To represent ambiguity and vagueness several codes can be submitted for one sentence. This way, player 1 can express what she thinks to be consistent with the feedback. In Kripke semantics this corresponds to what worlds she thinks to be possible given the feedback of player 2. The interpretation will not be visible to player 2.

Since the game is symmetric, player 1 also has to evaluate her own guess about the other player's code compared to her own secret code and this evaluation is checked by an algorithm and is not visible to player 2. Player 1 then has to provide feedback to player 2 about her own guess compared to her secret code. This will be done in the same way as player 2 did, by communicating sentences. Player 2 has to submit his interpretation of the feedback sentences he gets from player 1, which will not be visible to player 1. This ends the turn so that now player 2 can make a guess.

Here follows an example. Suppose Mary has chosen her secret code to be red, blue, green, yellow and John's secret code is orange, purple, blue, green. It is John's turn to make a guess and he guesses green, orange, brown, yellow. Now, Mary and John have to give an evaluation of the guess. Compared to Mary's secret code, there are two correct colors (green and yellow) and one color is in the right place (yellow). So Mary's evaluation is black, white. Similarly, John's evaluation is white, white (from orange and green). Although Mary and John have to submit these evaluations, they cannot see each other's evaluation.

Now Mary and John have to provide feedback sentences to each other. Mary chooses to communicate the sentences 'Sommige kleuren zijn goed.' (*Some colors are right.*) and 'Een kleur staat op de goede plaats.' (*There is a color which is in the right place.*). She sends these sentences to John. At the same time, John selects the sentences 'Twee kleuren zijn goed.' (*Two colors are right.*) and 'Geen kleur staat op de goede plaats.' (*No color is in the right place.*), and sends them to Mary.

After receiving the feedback sentences, both players need to submit their interpretation of the feedback sentences they received. Mary thus needs to submit what worlds she holds possible given that two colors are right and no color is in the right place. She could submit the combination (white, white). Submitting only this combination would mean she has a pragmatic interpretation of 'twee' (*two*). She could also submit a logical interpretation by adding (white, white, white) and (white, white, white, white), allowing 'twee' to mean *at least two*.

John has to give his interpretation of the feedback sentences he got from Mary. So John has to tell what worlds he considers possible given that some colors are right and there is a color which is in the right place. He could submit (white, black). If he would only submit this combination, his interpretation would be pragmatic with 'sommige' (*some*) meaning *exactly two* and 'een' (*a*) meaning *exactly one*. He could also submit (white, white, black), allowing 'sommige' to mean *exactly three*, or (white, black, black), allowing 'sommige' to mean *exactly three* and 'een' (*a*) to mean *exactly two*. If he includes one of the last combinations, his interpretation would be more logical.

After both players have submitted their interpretation the turn ends and the other player, in this case Mary, can make a guess in the following turn. Not only in interpretation, but also in production (choosing feedback sentences) can players be more or less pragmatic. In the above example, Mary could have been less pragmatic in production by choosing the sentence 'Een kleur is goed.' (*There is a right color.*) instead of 'Sommige kleuren zijn goed.'. Similarly, John could have preferred to use 'Eén kleur is goed.' (*One color is right.*).

Motivation of choices in game design

For the secret code, four different colors have to be chosen out of seven colors, and their order must be determined. There are 840 ($7 \times 6 \times 5 \times 4$) codes possible, given these restrictions. There are 35 ($7! \,/\, 4! \times (7-4)!$) possible combinations of colors. Once it is known which four colors the code is made up of, there are still 24 possibilities ($4 \times 3 \times 2 \times 1$). It is difficult to predict how many guesses will be needed to guess the secret code, because this strongly depends on the feedback given and the interpretation of the feedback. To prevent the game from taking too long and participants from getting bored, each participant can guess at most eight times per game.

A total of seven colors will allow for wrong colors, but keeps the total number of possible combinations reasonable. With fewer than seven colors, at least two colors would always be right and thus sentences about color with 'één' (*one*) or 'een' (*a*) would never be true. To further restrict the number of possible combinations, the additional restriction that the code must contain four different colors has been made. Without this restriction, there would be 2401 (7^4) possibilities. This restriction also prevents the reasoning aspect of the game from becoming too difficult.

A code of four colors is chosen because this is the shortest number which enables interesting possibilities in language use. Consider the following example.

1. Sommige kleuren zijn goed.

   Some colors are right.

2. De meeste kleuren zijn goed.

   Most colors are right.

3. Alle kleuren zijn goed.

   All colors are right.

4. Een kleur is goed.

   There is a right color.

When from these available sentences, 1 is given as feedback, the pragmatic interpretation following Grice's quantity maxim (see section 2.4) would be that two colors are right. This is because if three colors were right, it would be more informative to use 2, when four were right 3 and when only one was right 4. In a logical interpretation on the other hand, there could be one, two, three or four correct colors. So using a four color secret code, the interpretations the players submit will reveal whether they have a fully pragmatic or a more logical interpretation. When communicating a sentence, a player's intended meaning can be derived from the evaluation of the guess. This will reveal whether a pragmatic or logical meaning has been intended in production.

If fewer than four colors would be used, the strictly pragmatic use of 'sommige' (*some*) and 'de meeste' (*most*) would not be distinct. Thus, less variation in interpretation and production would be possible. With four items, you could allow 'sommige' to mean *exactly two* or *exactly three*, but not *exactly one* or *exactly four*. Perhaps the difference in use between 'sommige' and 'de meeste' is maintained less strictly by some people than the difference between 'sommige' and 'alle' (*all*). With fewer than four colors this cannot be tested. It is also slightly counterintuitive to speak of 'de meeste' or 'sommige' when there are only three items. This would more readily invite participants to question the meaning of the scalar terms and thus have an unwanted influence.

Because at least four different items are needed, a guess has to be a possible secret code. So for example, guessing four times red is not allowed. When a guess has fewer than four different colors, this can be seen as having fewer than four items in the color dimension.

It can be a bit impractical for participants to have to submit their interpretation of the feedback as described. If the following sentences would be given:

- Sommige kleuren zijn goed,

  Some colors are right,

- Sommige kleuren staan op de goede plaats.

  Some colors are in place.

the logical interpretation could only be expressed by ten different four item codes. It would be easier if the participants could just select the codes which they think possible. The reason this is not done is that this could make them consider options they would normally not think of.

To make sure participants evaluate their opponent's guess correctly, their evaluation (not their interpretation) is checked by an algorithm. This is because if incorrect feedback is given as a result of an incorrect evaluation, one player is misled in a way that is not according to the game rules and the game gets seriously disturbed.

Available feedback sentences

The feedback sentences from which participants can choose are listed in Appendix B.1. A translation into English is provided in B.2. Two scales for scalar terms are used: ⟨1, 2, 3, 4⟩, ⟨geen, een, sommige, de meeste, alle⟩ (⟨*no*, (*there is*) *a*, *some*, *most*, *all*⟩). The same feedback sentences are available throughout the game.

Many other interesting scalar terms could be investigated such as 'ten minste' (*at least*), 'hoogstens' (*at most*), 'minder dan' (*less than/ under*), 'meer dan' (*more than/ over*). One could also look at combinations of scalar terms and negation. However, it is not very practical to include this all at the same time. Since this is a pilot study, the number of scalar terms used is kept small.

4.2 Design

Consider a player with a strictly pragmatic language use. To this player 'één' means *exactly one*, 'twee' means *exactly two*, etc. This is because Grice's quantity maxim would otherwise be violated. As explained in the previous subsection, 'een' means *exactly one*, 'sommige' means *exactly two*, 'de meeste' means *exactly three* and 'alle' means *exactly four*, for this player in the situation provided by the game. 'Geen' means *none*, because feedback sentences have to be true in the game.

This player will choose feedback sentences of the same pragmatic strength, regardless of whether he wants to be as informative or as uninformative as possible. There is no way for him to reveal less information, because it would imply lying.

This does not hold for a player with a strictly logical language use. To this player 'één' will mean *at least one*, 'twee' will mean *at least two*, etc. 'Een' and 'sommige' will both mean *at least one*, 'de meeste' will mean *at least three* and 'alle' will mean *at least four*.

1. Eén kleur is goed.

   One color is right.

2. Twee kleuren zijn goed.

   Two colors are right.

If the logical player uses a strategy of being uninformative, he will always favor sentence 1 over sentence 2. This is because, according to this player, 1 is true in all cases where 2 holds and in some cases where 2 does not hold as well. If the logical player wants to be informative, he will therefore favor 2 over 1. In analogy, a sentence with 'een' will be favored over a sentence with 'de meeste' by a logical player who wants to be uninformative and vice versa for a logical player being informative.

Table 4.1 shows the different combinations of strategy and interpretation described and the expected behavior in feedback sentence selection. 'a' means preferring the sentence which is optimal according to Grice's quantity maxim. 'b' means favoring a less informative sentence over a more informative sentence.

Table 4.1: Possible combinations of strategy and interpretation.
a means favoring informative sentences.
b means favoring uninformative sentences.

strategy / language use	pragmatic	logical
being informative	a	a
being uninformative	a	b

Ideally, table 4.1 would contain 'a', 'b', 'c', 'd'. This would enable differentiation between all combinations based just on the feedback sentences selected. An assumption made, is that players start with a pragmatic interpretation and a poor strategy. This will result in behavior *a*. A shift to behavior *b* can only take place if the interpretation of the player becomes more logical and he has developed a strategy of revealing fewer information.

A player's way of interpreting will be measured through the interpretations he provides of the feedback sentences that he has received. Production will be measured through feedback sentence selection. Strategy will be determined by the answers to the questions the player is asked throughout the game, about why he makes certain choices.

Altogether, the experiment can reveal whether or not language use shifts from pragmatic to more logical and whether a strategy of being uninformative is developed.

4.3 Predictions

Grice's maxims are best applied in situations where conversation is cooperative. Since a rational strategy for playing the game in the experiment is to be as uninformative as possible conversation will probably not be cooperative in the experimental conditions. So once the players have mastered the game well enough to think about strategy and have become familiar with the uncooperative context, they are expected to develop a less pragmatic use of the sentences. How pragmatic or how logical language use is, may differ for each scalar term. There might also be an asymmetry between production and interpretation, as with children.

Since the game involves quite a lot of actions which need to be performed each turn, players are expected to start with a very simple or no strategy. As they get more experienced in playing the game they will have enough resources left to develop a more complex strategy. The following development is expected:

Stage 0 The player will choose to communicate sentences which are true and make a guess based on the feedback sentences he gets. The interpretation of the feedback sentences might be logical (because no capacity is left to draw pragmatic inferences) or pragmatic (because drawing pragmatic inferences is an automatic process). In the first case hypothesis 3 does not hold.

Stage 1 The player will explicitly take into account what his opponent knows, wants and believes (first order ToM). He will make guesses based on what he wants to know and what he knows from the feedback he got about his own and the opponent's preceding guesses.

When choosing a sentence to communicate, he will consider the amount of information that is revealed. This may result in less pragmatic productions. Eventually, when making a guess, he will ensure that he does not have to reveal too much information to the opponent as a result of his guess.

The interpretation of the feedback sentences the player gets will be pragmatic, since he considers the alternatives his opponent has in choosing a sentence.

Stage 2 The player will take into account what his opponent does not want him to know (second order ToM). He will be aware that his opponent is trying to reveal little information. This will lead to a more logical interpretation. The player will also try to infer information from the choices his opponent makes in guessing. For example, when his opponent chooses a certain guess over a more obvious one, he will check if the more obvious guess could reveal something his opponent would not want him to know.

Stage 3 The player will take into account that his opponent knows that he does not want his opponent to know certain things (third order ToM). He will also be aware that his opponent knows that he knows that his opponent does not want to reveal certain information (third order ToM). He may consider possibilities to mislead his opponent, using this knowledge. Since possibilities to do so are very limited due to the symmetric nature of the game, the development of strategy will stop at this stage for most players. Language use will be at its most logical end for this player. The improvement the player makes will be in reducing errors, becoming faster and having to put less effort into playing the game.

The reasoning about not having to reveal too much information to the opponent as a result of a player's own guess, in stage one, will only occur after the reasoning about the available information, which is needed for a guess, is sufficiently mastered.

The stages are based on the development of a theory of mind of the opponent. The number of the stage corresponds to the order of the ToM needed for the strategy described. Because less pragmatic productions can result from first order ToM use whereas less pragmatic interpretation requires second order ToM use, more logical production is expected to precede more logical interpretation. Individuals may differ in the speed in which they cross the stages and in the stage they eventually reach. The experiment may not allow for enough training to reach stage 2 and 3. Individual differences in how logical language use becomes are expected to occur.

The logical use players eventually reach, results from a conscious reasoning process (hypothesis 3). This means players will be able to describe this part of their strategy.

Due to the improved strategy trained players will outperform novices. If this is not the case, either the trained player has not learned to use his theory of mind effectively or the novice can already do so.

4.4 Procedure

At the first session, participants were first given a written instruction (see appendix C) on how to play the game and how to use the computer program. They were allowed to ask in advance any questions considering the game rules or the computer program. In addition to the written instruction an oral instruction was provided by the experimenter. This instruction included that the participants were not allowed to communicate any information on the experiment to prospective participants, talk to their opponents, ask questions (except questions on the use of the computer program), think aloud, tell the opponent their secret code if it wasn't guessed (this to prevent people from suspecting foul play in case of differing interpretations).

After the instructions each participant played the game against another participant for approximately three hours. During one of these games, after an hour and a half had passed, participants needed to answer a question on why they chose a particular guess and on why they selected particular feedback sentences. After three hours, participants filled in questionnaire one (see appendix D), which was on their strategy.

At the second session, participants were allowed to read the instructions again and were given the opportunity to ask questions. They then played the game again for three hours in the same way as in session one. After playing the game, they completed questionnaire two, which was the same as questionnaire one, and questionnaire three (see appendix D), which was on their background and prior knowledge.

Participants did not receive any reward for winning the game nor a punishment for losing. They were not allowed to write anything down during the game. The time between the sessions differed between participants because of practical limitations.

4.5 Materials

To enable participants to play the game via computers, a network was established of three Celeron processors of 766 MHz. One desktop was used for each player and one desktop was used as a server. The operating system used was Red Hat 9. The implementation of the game Master(s)Mind(s) has been developed in Kylix 3.

4.6 Participants

Nine people (7 male, 2 female) completed the two session experiment. Their ages were 21, 22, 22, 22, 23, 23, 26, 26, 60. Three people (2 male, 1 female) completed only one session, their ages were 22, 24, 58. All participants volunteered for the experiment, one of them was rewarded with a kind of credits towards a degree in Psychology. This participant played the second session against the experimenter. Table 4.3 and 4.2 show the relevant knowledge and experience participants had as measured via questionnaire 3.

Table 4.2: Relevant experience of the participants.
The numbers represent the participants.

subject	No experience	Some experience	Fair amount of experience	A lot of experience
working with computers			3, 5, 6, 7, 10, 12	1, 2, 4, 8, 9, 11
playing strategic games		3, 6, 7, 8, 12	1, 2, 4, 5, 9, 10	11
playing mastermind	3	2, 4, 5, 6, 7, 8, 9, 11, 12	1, 10	
playing games similar to mastermind	1, 3, 4, 5, 9, 11	2	10	

Table 4.3: Background knowledge of the participants.
The numbers represent the participants.

subject	hardly any knowledge	some knowledge	fair knowledge	a lot of knowledge
logic	6, 8, 11, 12	5	1, 2, 3, 4, 7, 9, 10	
epistemic logic	3, 5, 6, 7, 8, 9, 11, 12	1, 2, 4, 10		
theory of mind	1, 2, 3, 5, 6, 7, 8, 9, 11, 12	4		10
truth conditional semantics	3, 5, 6, 7, 8, 9, 10, 11, 12	1, 2, 4		
pragmatics	2, 3, 5, 7, 8, 9, 11, 12	1, 4, 6, 10		

Chapter 5

Results

This chapter starts with a description of the results considering language use. Then the results regarding strategy are given. Third, the results about the order of theory of mind the participants have used are presented. The section ends with the results about what choices participants made in choosing feedback sentences.

The participants are numbered from one to twelve, the participants 10, 11, and 12 completed only one session.

5.1 Language use

Tables 5.1 and 5.2 show the initial and final language use of the participants. This is the language use as measured in interpretation. Most participants showed symmetry between interpretation and production. Participant 3 however, stayed more logical in production. Participant 5 allowed 'sommige' (*some*) to mean *exactly one* in production, but not in interpretation. Participant 9's interpretation eventually became pragmatic, but his productions did not. Table 5.3 shows the initial and final language use of each participant by category. The following shifts in interpretation and production occurred:

een Participant 3 shifted to a more pragmatic language use of 'een' (*a*) over the course of the experiment. Participants 2, 4, and 5 shifted to a more logical use of 'een'.

sommige Participants 1, 2, 3, 4, 6, 7, and 12 (seven in total) shifted to a more pragmatic use of 'sommige' (*some*). Participant 9 shifted to a more pragmatic interpretation only and was constant in the production of 'sommige'.

de meeste Participant 10 shifted to a more pragmatic use of 'de meeste' (*most*), whereas participant 4 shifted to a more logical use of 'de meeste'.

één Participant 3 shifted to a more pragmatic use of 'één' (*one*). Participant 4 and 5 shifted to a more logical use of 'één'.

twee Participant 3 shifted to a more pragmatic interpretation of 'twee' (*two*).

drie Participant 4 shifted to a more logical use of 'drie' (*three*).

Table 5.1: Initial and final interpretation of quantifiers.
The boldfaced numbers between brackets, above the columns, represent the meanings that participants included. '1' should be read as *exactly one* etc. Thus, **(1,2)** means the interpretation that the given scalar can mean *exactly one* and *exactly two*. The numbers in the rows represent the participants, so '1' should be read as *participant one* etc. This way, the table shows which participants had a certain interpretation at the beginning and at the end of the experiment, for each scalar term used. The numbers of the participants that changed their interpretation for a certain scalar term are in italic in the row with the final interpretations. The second column contains the pragmatic interpretations, which are a subset of the logical interpretations. For 'geen' and 'all', the pragmatic and logical interpretation are identical.

geen	**(0)**			
initial	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12			
final	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12			
een	**(1)**	**(1, 2)**	**(1, 2, 3)**	**(1, 2, 3, 4)**
initial	2, 4, 5, 6, 7, 8, 9, 10			1, 3, 11
final	6, 7, 8, 9, 10	*2*	*3*, *5*	1, *4*, 11
sommige	**(2)**	**(2, 3)**	**(2, 3, 4)**	**(1, 2, 3, 4)**
initial	8	5, 6, 7, 9, 10, 12	4, 11	1, 2, 3
final	*6*, *7*, 8, *9*, 12	*3*, *4*, 5, 10	*1*, *2*, 11	
de meeste	**(3)**	**(2, 3)**	**(3, 4)**	**(2, 3, 4)**
initial	5, 6, 7, 8, 9	4	1, 2, 3, 10, 11	
final	*4*, 5, 6, 7, 8, 9, *10*		1, 2, 11	*3*
alle	**(4)**			
initial	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12			
final	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12			

Table 5.2: Initial and final interpretation of numbers.
This table is similar to table 5.1, thus the boldfaced numbers between brackets, above the columns, represent the meanings that participants included, and the numbers in the rows represent the participants. The numbers of the participants that changed their interpretation of a certain scalar term are in italic in the row with the final interpretations. The second column contains the pragmatic interpretations, which are a subset of the logical interpretations. For 'vier', the pragmatic and logical interpretation are identical.

één	**(1)**	**(1, 2, 3)**	**(1, 2, 3, 4)**
initial	1, 2, 4, 5, 6, 7, 8, 9, 10, 12	3	
final	1, 2, *3*, 5, 6, 7, 8, 9, 10, 12		*4*
twee	**(2)**	**(2, 3, 4)**	
initial	1, 2, 5, 6, 7, 8, 9, 10, 12	*3*, **4**	
final	1, 2, 3, 5, 6, 7, 8, 9, 10, 12	4	
drie	**(3)**	**(3, 4)**	
initial	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 12		
final	1, 2, 3, 5, 6, 7, 8, 9, 10, 12	*4*	
vier	**(4)**		
initial	1, 2, 4, 6, 7, 8, 9, 10, 12		
final	1, 2, 4, 6, 7, 8, 9, 10, 12		

Table 5.3: Language use.
This table shows the type of language use (pragmatic or logic) of participants during the experiment, initially and finally. The numbers represent the participants. The numbers of the participants who made a shift are in italic in the row that represents the final language use.

	pragmatic	fairly pragmatic	fairly logical	logical
initially	8	5, 6, 7, 9, 10, 12	1, 2, 3, 4	11
finally	*6, 7, 8, 12*	9, 10	1, 2, 3, 4, *5*	11

Participant 8 had a fully pragmatic language use. Participants 7, 9, 10 and 12 started with a fairly pragmatic use which became even more pragmatic during the experiment. Participants 7, 8, and 12 eventually had a fully pragmatic language use.

Participant 5 started with a fairly pragmatic use, but shifted to a somewhat more logical use during the second session of the experiment. In this session he was playing against a participant with a fully logical use.

Participant 11 started with a fully logical language use and kept this use throughout the experiment. Participants 1 and 2 showed a fairly logical use. Participant 4 shifted from a language use in between logical and pragmatic to a more pragmatic use in the first session (which was against a participant with a fairly pragmatic use). In the second session he started with a fairly logical use. This session was against a player with a fairly logical use whom he knew. Participant 3 started with an almost fully logical use, and shifted towards a more pragmatic interpretation (he was playing against opponents with a fairly logical use). Although his interpretation of the quantifiers stayed fairly logical, his interpretation of the numbers shifted to fully pragmatic.

Participant 1 wrote that 'één' (*one*) could be used in a situation where two holds, but that he did not consider this a natural use. He wrote that he noticed that his opponent acted accordingly. He also wrote that he thought that the sentence 'een kleur is goed' (*there is a right color*) was always true, but that he didn't have the guts to start an argument on this. Some other participants wrote explicitly on what a term could or could not mean as well.

Some participants thought that their opponent was playing foul by lying.

5.2 Strategy

A participant's strategy was measured by the answers the participant gave to the questions during the game and on the questionnaires. For example, participant 8 wrote that he tried to avoid ambiguity when selecting feedback sentences, whereas participant 3 wrote that he tried to give as little information as possible when choosing sentences. Table 5.4 is an overview of the strategies used by the participants and the changes in strategy that occurred.

Table 5.4: Strategy.
This table shows what kind of strategy participants used during the experiment, initially and finally. The numbers of the participants who made a shift are in italic in the row that represents the final strategy.

	being uninformative	being informative	other
initially	1, 2, 4, 5, 10, 11	3, 8, 9, 12	6, 7
finally	1, 2, *3*, 4, 5, 11	9, 12	*2*, *3*, 6, 7, *8*, *10*

Four players (1, 4, 5, 11) started out with a strategy of revealing little or no information and consistently used this strategy throughout the experiment.

One player (10) started out with the strategy of revealing little information, but changed this strategy to not making things too difficult for the opponent. One player (2) started with the strategy of revealing little information and changed this into not revealing too much information as long as little was known about the opponent's secret code. One player (7) just alternated between being clear and being vague.

Two players (9, 12) wanted to be as clear as possible, or didn't want to make things too difficult for their opponent and stayed with this strategy.

One player (3) started with the intention of being clear and giving as much information as possible, then shifted to revealing very little information and eventually tried to confuse the opponent.

One player (6) wanted to make things difficult for the opponent, but did not clearly relate this to the amount of information given. One player (8) started with being as informative as possible and changed this to making things difficult, but also did not relate difficulty to the amount of information being revealed.

In guessing the secret code, a number of players first concentrated on getting all of the colors right. Participant 4 wrote that he focused on what was wrong, rather than on what was right. A lot of players did not consider the amount of information that was revealed by, or had to be revealed as a result of their own guesses.

Player 9 tried to make his opponent lose his overview by using the quantifier scale. In the overview of the computer program, abbreviations were used for the feedback sentences. He wrote that whereas K4P1 (meaning four colors are right, one color is in the right place) was immediately clear to him, KaPe (meaning all colors are right, there is a color which is in the right place) first needed translation into K4P1.

Player 8 at some point concluded that using the number scale might result in more difficulties for his opponent, because it seemed less friendly.

Player 2 wrote that he tried to disturb his opponent by giving the same feedback sentences repeatedly.

Social factors had an influence on the strategy players used. Some players changed their strategy because they wanted to give their opponents a better chance of winning the game, after they had won several games in a row. As mentioned, participant 1 wanted to avoid a conflict, which led to revealing more information than necessary according to his own interpretation.

It was not found that players who applied second order theory of mind, or had a strategy of revealing little information, had a better chance of winning the game. Trained players did not outperform novice players consistently.

5.3 Order of Theory of Mind used

The order of the theory of mind that was used by a participant was determined by analyzing the written remarks the participant made and the answers given to the questions. Table 5.5 shows the highest order of ToM used by each participant.

The order of the theory of mind that was used by a participant was determined by analyzing the written remarks the participant made and the answers given to the questions. Table 5.5 shows the highest order of ToM used by each participant.

Participant 1 showed the use of a second order ToM halfway the first session. He wrote that in making a guess, he considered that the guesses made by his opponent were evasive from his opponent's secret code, since he wanted to hide

Table 5.5: Highest Order of ToM used.
This table shows the highest order of ToM that participants used during the experiment. The numbers represent the participants.

1st order	possibly 2nd order	2nd order
3, 5, 6, 7, 8, 9, 10, 12	4	1, 2, 11

this code. So he used a second order thought like 'he does not want me to know' to analyze the behavior of his opponent and used the information he gained this way to guide his own behavior. Participant 1 also wrote that he tried to make guesses which did not resemble his own code too much nor too little, he preferred some colors to be right. This indicates the use of first order ToM.

Participant 2 wrote that he thought that red should be in the secret code of the opponent, because his opponent always included red in a guess to make sure that some colors were right. Although it cannot really be concluded from this remark, this strongly points in the direction of the use of second order ToM. The reasoning on which this remark is based is probably similar to the following:

> My opponent does not want me to know anything about his secret code (second order ToM). My opponent knows that if he can choose 'sommige kleuren zijn goed' (*some colors are right*) this desire is fulfilled as much as possible, therefore he will include at least two colors of his own secret code. Since red is always included in his guesses, he might use this color to make sure that two colors are right. It is therefore that red must be in his secret code.

In participant 2's interpretation, if only one color would be right his opponent would have to reveal three colors of his secret code. This is because there are only seven colors to choose from, 'een' (*a*) and 'één' (*one*) both mean exactly one, and no other expression to choose from can mean one. It is only initially that participant 2 allows 'sommige' to mean one. Apparently, participant 2 assumes his opponent to have this first order knowledge (if I do such and such, then my opponent will know ...). This also indicates the use of second order ToM.

Participant 4 wrote that he thought his opponent in the second session played the game well, because he sometimes used 'één' (*one*) in cases where *exactly two* held. It seems plausible that he assumed his opponent to know that by doing so, he was revealing little information. If so, he attributed first order knowledge to his opponent and thus had at least a second order theory of mind of his opponent. However, it is unclear whether participant 4 applied his second order ToM, because his own interpretation was very logical right from the start of the second session and was not influenced by the productions or guesses of his opponent. There were no other remarks that confirmed the application of second order ToM for participant 4. Participant 4 knew that his opponent studied mathematics, which may have caused him to expect more logical productions and thus may have led to a logical interpretation. In this case, the logical interpretation would result from first order ToM use.

Participant 4 was using first order ToM. This is clear because he wrote that in choosing feedback sentences, he considered what his opponent already knew.

In his first session, he also wrote that he did not make guesses in which only one of the colors of his own secret code was included.

Participant 11 wrote that he made a certain guess because he wanted to check whether the guesses his opponent made had anything to do with his opponent's secret code. This could result from a second order thought like: 'he could be guessing his own secret code, because he knows that I would not know that his guess is his secret code.' Participant 11 frequently used his own secret code as a guess. Since his interpretation was very logical, this did not force him to reveal much information. Since he himself was aware that this didn't reveal much information, it could well be the case that he attributed this first order knowledge to his opponent and thus was using second order ToM.

The remaining participants (3, 5, 6, 7, 8, 9, 10, 12) showed no sign of second order ToM. All of them were using first order ToM though, because they all, at least occasionally, tried to make things difficult for their opponent or tried to reveal little information. None of the participants wrote down anything that could point to a third or higher order theory of mind.

As part of an answer to the first question of questionnaire one participant 9 wrote: 'Ik houd ook vrij weinig rekening met mijn eigen code, dwz om delen van mijn code te onthullen. Om te winnen richt ik me op de te kraken code en probeer niet mijn tegenstander te misleiden.' *(I also hardly consider my own code, that is to say to reveal parts of my code. To win I concentrate on the code to be guessed and I do not try to mislead my opponent.)* This demonstrates that participant 9 might have had some trouble using his first order theory of mind.

5.4 Behavior in choosing Feedback

As described earlier, depending on strategy and interpretation, two different types of behavior were expected. One for the players with a logical interpretation and a strategy of being uninformative and one for the players who did not have this combination. The behavior the logical, uninformative players can show as opposed to the other players is for example favoring sentence 1 over sentence 2 in the situation where two colors are right, or favoring sentence 3 over sentence 4 when all colors are right.

1. Eén kleur is goed.
   One color is right.
2. Twee kleuren zijn goed.
   Two colors are right.
3. Sommige kleuren zijn goed.
   Some colors are right.
4. Alle kleuren zijn goed.
   All colors are right.

Table 5.6 shows whether or not participants showed this type of behavior, and thus preferred less informative sentences over more informative sentences.

Table 5.6: The preference for uninformative sentences.
This table indicates which participants preferred less informative sentences, also see section 4.2. The numbers represent the participants. The numbers of the participants who made a shift are in italic in the row that represents the final behavior.

	preferred less informative sentences	did not prefer less informative sentences
initially	1, 3, 4, 5, 11	2, 6, 7, 8, 9, 10, 12
finally	1, 2, 3, 4, 5, 11	6, 7, 8, 9, 10, 12

Because most players did not have a strictly pragmatic or strictly logical interpretation, the player's own interpretation could be used to evaluate whether or not a less informative sentence was preferred over a more informative one. For example, a participant could have the interpretation that *de meeste* (*most*) can mean *exactly three* or *exactly four* and that *drie* (*three*) can only mean *exactly three*. If this player chooses a sentence containing *de meeste* in a case where exactly three colors are right, he could be said to prefer a less informative sentence (containing *de meeste*) over a more informative one (containing *three*). Using this definition would result in the same participants showing the behavior and the same participants not showing the behavior in this experiment.

Cases where the vaguer term was used in its preferred interpretation were not counted. Preferred interpretations were determined by determining what percentage of the participants initially included a certain interpretation (Table 5.7). The interpretation included by most participants (in bold) was assumed to be the preferred interpretation. In case of 'sommige' both two and three were counted as preferred interpretation, in fact, 'two or three' was the most popular initial interpretation of 'sommige'.

Some participants made mistakes in selecting feedback sentences, for example by choosing 'one' in case of *no* or 'all' in case of *exactly three*. Selections were considered a mistake if the feedback was not in accordance with the participant's interpretation and this particular choice of feedback for this situation did not occur consistently. Participant 5, 7, 8, 9, and 11 each made one mistake. Participant 6 made about one mistake every game. Some participants noticed their opponent's mistake(s).

It was not found that players who applied second order theory of mind, or had a strategy of revealing little information, had a better chance of winning the game.

Table 5.7: Preferred Interpretation

The columns indicate possible meanings of a scalar term; '1' should be read as *exactly one*, etc. The rows show what percentage of the participants of whom an interpretation was available for a certain scalar term, included a certain meaning for that scalar term. So of all participants who submitted an interpretation of 'een', everyone thought that '1' could mean *exactly one* and twenty-seven percent thought that '1' could also mean *exactly three*.

The boldfaced percentages indicate what interpretation is regarded as the preferred interpretation. For example, for 'een' this is *exactly one* and for 'sommige' this is *exactly two or exactly three*. To determine the preferred interpretation, the interpretations participants had at the start of the experiment were used. The interpretation of the vast majority was chosen as the preferred interpretation.

	0	1	2	3	4
geen	**100**	0	0	0	0
een	0	**100**	27	27	27
sommige	0	25	**100**	**92**	42
de meeste	0	0	9.1	**100**	45
alle	0	0	0	0	**100**
1	0	**100**	9.1	9.1	0
2	0	0	**100**	18	18
3	0	0	0	**100**	0
4	0	0	0	0	**100**

Chapter 6

Discussion

In the first section of this chapter, the results described in the previous chapter are compared to the hypotheses stated in chapter 3, and to the predictions described in chapter 4. In section 6.2, the results found in the Master(s)Mind(s)-experiment are compared to the results found by Feeney et al. [11], Keysar et al. [20], Papafragou and Musolino [30], and Noveck [29]. The final section of this chapter is on ideas for future work. These ideas are ordered per hypothesis to which they apply. In addition, a description of how cognitive modeling could be used in future work is given.

6.1 Hypotheses, Predictions and Results

Hypothesis 1

Hypothesis 1 stated that performing a task and simultaneously reflecting upon this task can be seen as a form of dual tasking. This led to the prediction that when playing Master(s)mind(s), participants would start with a simple strategy, since they had to concentrate on the first task, which is playing the game according to the rules. A simple strategy would lack the element of revealing little information. As the participants got more experienced, they would have more capacity left for the second task, developing a winning strategy, and thus shift to a better strategy which contains revealing little information.

Looking at the data found in the experiment, some evidence can be found for this hypothesis. Two participants changed their strategy of being informative during the game, but only one of them to being less informative. The other participant just tried to make things difficult for the opponent. However, this participant had a fully pragmatic language use and thus no means of being less informative, as explained in section 4.2. In this case, trying to make things difficult can therefore be seen as a better strategy.

Six participants did not use the strategy of being uninformative at all. It could be the case that they were still too much occupied with the first task. One of them made a lot of mistakes which indeed points in this direction. Three of them each made one mistake. Of the participants who used the better strategy, two participants made a mistake. Four of these six participants developed a fully pragmatic language use, and thus were not able to reveal less information through feedback sentence selection. The other two had a fairly pragmatic use.

Half of the participants almost immediately had a strategy of being uninformative. This is not in accordance with the predictions. However, it could be the case that the game was too easy for these participants, so that they could immediately do the second task as well. These participants made relatively few errors, only two of them each made one mistake, both of them during the first session. The participants with this advanced strategy had relatively much experience in using computers and playing strategic games and knew relatively a lot about logic. This background would certainly make the first task easier to them.

One participant changed strategy from being uninformative to informative, but this was because of social reasons.

Although the evidence for hypothesis 1 is far from convincing, there is no reason to abandon the hypothesis because of this experiment.

Hypothesis 2

Hypothesis 2 stated that in an uncooperative conversation, people will shift their language use from pragmatic to less pragmatic. Thus, it was predicted that while playing the game, the interpretation and production participants used would become more logical and less pragmatic.

Only two participants developed a more logical language use. Six participants on the other hand developed a (somewhat) more pragmatic use. Based on these data, hypothesis 2 should be abandoned. From table 5.1 on page 38 and table 5.2 on page 39, it can be seen that interpretations that were not pragmatic were used more frequently for the quantifiers than for the numbers. A possible explanation for this is that the numbers have the same pragmatic meaning in any situation, whereas the pragmatic interpretation of the quantifiers differs in different situations. For example, when six colors would have been used, the pragmatic meaning of 'sommige' (*some*) would haven been *exactly two or exactly three* instead of *exactly two*. For 'geen' and 'alle', there is only one meaning that is truth-conditionally true, and thus the pragmatic meaning is the same as the truth-conditional meaning. Although this explanation can account for the participants shifting to a more pragmatic interpretation, it does not explain the behavior of participants shifting to a less pragmatic interpretation.

None of the people with a pragmatic use developed the strategy of being uninformative. Thus, some people are still being informative in an uncooperative situation. For these people Grice's quantity maxim can be used to explain their language use, even in an uncooperative situation. Most of these people developed a more pragmatic use during the experiment.

It should be noted though, that six of the twelve participants eventually had a fairly logical, or logical use of language. It would be interesting to investigate whether these participants also have such logical language use in daily life. It seems plausible that the uncooperative situation influenced their language use.

All of the participants who eventually had a logical interpretation and production, eventually had the strategy of being uninformative.

Without the strategy of being uninformative, logical production would not be measurable, as explained in section 4.2, but logical interpretation would still be measurable, since participants gave their interpretation of the feedback they received. Thus, it can safely be concluded from the data that no participants without the strategy of being uninformative, had a logical interpretation.

On the other hand, being uninformative required a more logical use. The only thing a participant with a pragmatic use could do to be less informative, is to carefully choose his own guesses such that he did not have to say too much about his own code. In choosing feedback sentences, he could not be less informative, because according to his explanation this would imply lying (also see section 4.2).

Being uninformative is only useful in uncooperative conversation. The participants with a logical language use had the strategy of being uninformative, and thus were aware of the uncooperative context. So, although hypothesis 2 does not hold, the situation of conversation being uncooperative may still influence language use, at least for some people.

Hypothesis 3

Hypothesis 3 stated that the pragmatic use of quantifiers is the result of an automated process, which can be overruled by a deliberate reasoning process, which would result in a more logical use of the quantifiers.

Four people started out with a fairly pragmatic use, which became more pragmatic during the experiment (participants 6, 7, 9, and 12). Only one participant started with a fully pragmatic use. To these people developing a more pragmatic interpretation was a benefit, as long as their opponent had the same kind of production, because they would gain more information from a more pragmatic interpretation than from a less pragmatic interpretation. These data suggest that the pragmatic use is not a fully automated process in any situation. Thus, hypothesis 3 should be abandoned.

One participant shifted from a fairly pragmatic to a fairly logical use during the experiment. This may have been the result of the behavior of his opponent, who consistently selected the sentence 'een kleur is goed' (*there is a right color*). This participant show that the pragmatic interpretation can be overruled.

One participant literally wrote down that he thought that 'een kleur is goed' was always true, but that he did not consider this a natural interpretation. It seems that this participant made the deliberate reasoning process which resulted in a more logical interpretation, even though he did not dare to use this in his productions.

The utterances participants wrote down about the interpretation of the scalar terms indicate that deliberate reasoning about language use took place during the experiment.

Again, it would be interesting to know whether the participants with a logical use, have a more pragmatic use in daily life. This certainly seems plausible and if so, they could be said to overrule this use. However, since they were doing so from the start, it seems that they do not have to put a lot of effort into overruling a pragmatic use.

Predicted stages

A difference between the predicted stages and the experimental results is that participants do not show all the reasoning possible and useful with first order of theory of mind. Although all participants used first order ToM, not all of them considered the amount of information they were revealing.

From the predicted stages, an asymmetry between production and interpretation was expected, because more logical productions can result from first order ToM use, whereas more logical interpretations result from second order ToM use. Thus, a shift in production is expected to precede a shift in interpretation.

This would be particularly relevant for participant 5, since this participant shifted to a more logical language use. However, this participant's more logical interpretation preceded his more logical productions for 'een' (*a*). Participant 5 was playing against participant 11 in his last session. Participant 11 almost solely gave feedback sentences containing 'een' (*a*). It seems that after participant 5 understood the way participant 11 was using 'een', and thus changed his interpretation, he started using 'een' in the same way, and thus changed his production. In this case, the more logical interpretation did not result from second order ToM use, but from reasoning about the colors in the codes guessed, the feedback received and the rules of the game. For example, when having used all seven colors and only having received the feedback that a color was right in each code, one can infer that 'a color' cannot mean *exactly one color*.

Participant 5 allowed 'sommige' *some* to mean *exactly one* in production, and not in interpretation. Here production may be preceding interpretation, but it seems unlikely since this type of productions occurred from the start of the experiment, and were not followed by similar interpretations.

Participant 3 and 9 changed their interpretation more than their production. The other participants showed symmetry in production and interpretation, including the participants who shifted to a more pragmatic language use.

All of the participants with a second order ToM showed a logical language use. This is in line with the predictions. Not all participants who were using at most first order ToM had a pragmatic use, although six out of eight did. For participant 5 this could be explained by assuming he was copying his opponent's behavior. It might be the case that participant 3 was using a higher order ToM, but this could not be measured because he did not write down any second order thoughts. It could also be the case that the logical use of participant 3 results from experience and familiarity with logics, since he is a math student. Participant 3 and 5 were using a strategy of revealing little information and had a logical use, just like the participants who showed signs of second order ToM.

Probably four participants started in stage 2. Of one of them it cannot be concluded that he was using second order ToM, although this seems plausible. A lot of participants did not show all characteristics of stage 1.

It does not seem that participants made transitions between the stages during the experiment. Instead of developing themselves in the game, most of them got tired. The result that trained players did not outperform novice players is in line with this. The result that participants with more advanced strategies did not have a significantly better chance of winning the game than players with less advanced strategies could be because they also had to play against each other.

It is clear that some participants were using second order theory of mind and complex skills and strategies as mentioned in the research question. However, it does not seem that they acquired any of these skills while repeatedly playing the game Master(s)Mind(s). They took the skills to the start of the experiment. Though some transitions in language use and strategy took place, most participants were rather constant in their behavior.

6.2 Comparison with other experimental results

Feeney et al.

When having to judge whether infelicitous sentences containing *some*, such as 'Some elephants have trunks.' were either true or false, Feeney at al [11] found that three participants (6.2%) gave pragmatic responses only, twenty-five (51%) gave logical responses only and twenty-one (43%) gave a mixture of pragmatic and logical responses. In this example, the pragmatic response would be *false* whereas the logical response would be *true*.

In the Master(s)Mind(s)-experiment only one participant (8.3%) was fully pragmatic in language use, one (8.3%) was fully logical, and ten participants (83%) showed a mixture. During the experiment, three more participants (25%) became fully pragmatic in their language use instead of using a mixture.

The number of people who might not be capable of a logical language use is very low, and quite comparable in both experiments. However, the number of people who had a fully logical language use is quite different. This may be because of the different natures of the experiments, the fact that in the Master(s)Mind(s) experiment more scalar terms were used, or because of differences between English and Dutch. When considering the people capable of almost fully or fully logical language use in the Master(s)Mind(s)-experiment, the numbers are far more alike: 51 % vs. 50%. Both experiments make it clear that not all adults have to use pragmatic language use.

Feeney at al neither created a situation where language use would be cooperative nor where it would be uncooperative. In the Master(s)Mind(s) experiment, the conversation was meant to be uncooperative. Still, little difference was found in the language use of participants in both experiments, and if any it would be that participants were more pragmatic in their language use in the Master(s)Mind(s)-experiment, as opposed to the predictions.

The assumption made in this study, that the pragmatic language use would be overruled because of an uncooperative situation does not explain these data well. Feeney et al. do not specify under what conditions the overruling takes place. Clearly, this is something future work should address.

Keysar et al.

Keysar, Lin and Barr [20] found that adults do not reliably use their first order theory of mind to interpret the actions of others, nor to base their own actions on (also see section 2.1). In the Master(s)Mind(s)-experiment it was found that although participants used first order ToM, they did not use all kinds of reasoning possible, and useful in this context, with a first order ToM.

Most participants were aware of the desire of their opponent to know their secret code and wanted to make it as difficult as possible for the opponent to get that knowledge. Most of them where considering how their opponent would interpret a certain feedback sentence. However, few participants considered what information would be revealed or have to be revealed as a result of a certain guess they made themselves. Apparently, participants were more aware of information revealed by their language use than they were of information revealed by their actions.

Participants who used second order ToM did use all the reasoning patterns with first order ToM which were described above. Thus, they considered guesses as well as sentences to be informative. In the experiments of Keysar et al., participants had to produce descriptions or follow instructions in natural language. Thus, the participants also made mistakes in their language use as a result of a poor theory of mind use.

A comparison between the results of Keysar et al. and the results of the Master(s)Mind(s)-experiment is that in both experiments some people did not fully use their first order ToM to guide their actions. A difference however is, that in Master(s)Mind(s), the mistakes were not so much in language use, but in another aspect of the game.

Papafragou & Musolino, Noveck

In [30], Papafragou and Musolino present results that indicate that adults reject most (over 90 %) infelicitous sentences containing *some* instead of *all*, *two* instead of *three*, or *start* instead of *finish*. Their experiment was in Greek. In their experiment, when adults were asked whether a puppet had 'answered well' to a question asking what happened in a story, they preferred the pragmatic interpretation in over ninety percent of the cases. In [29], Noveck presents similar results for the English scale ⟨ *has to be, does not have to be, might be, cannot be*⟩ and the French scale ⟨*Certains (Some), Tous (All)* ⟩.

Although the results from the Master(s)Mind(s)-experiment do not contradict these results, they do suggest that not all adults would respond in the same way to similar experiments. One participant in the Master(s)Mind(s)-experiment never used a pragmatic interpretation or production, whereas another participant always did so. Although the other participants were in between these extremes, there clearly was a group who most often had a logical language use and a group who most often had a pragmatic language use.

In addition, the Master(s)Mind(s) experiment indicates, just as the experiment from Feeney et al. described above, that there is no clear point from which people prefer a pragmatic interpretation. Feeney et al. and Papafragou & Musolino showed that children's performance in the pragmatic interpretation of scalar terms, depends on the context. The Master(s)Mind(s)-experiment and the experiment of Feeney et al. show, that under certain conditions, adults will prefer a logical language use above a pragmatic language use. More research will be required to further specify these conditions, make it more clear how many stages there are in scalar term interpretation, and get a better view of individual differences.

6.3 Future Work

Hypothesis 1

In future work, more evidence for or against hypothesis 1 has to be found. To exclude the possibility that the first task is just too hard or too easy for some participants, the difficulty of this task needs to be varied. In the Master(s)Mind(s)-experiment, there are several ways to do so.

The interface of the computer program could be made less user friendly. In the current implementation, there is a window in the lower-left corner which always shows an instruction to the participant about what needs to be done next. This could clearly reduce the cognitive load of the first task. There are other instructions shown as well, during the game. Although some participants would not be able to play the game without this help, for some participants this may be an adjustments which just makes it a bit harder to concentrate on the complex skills.

Another way to increase the difficulty of the game is to add time pressure. This would give participants less time to reason about the available information, which is the first task, and thus leave even less time for the second task, developing a strategy and reasoning from implicated meaning.

Difficulty could also be increased or reduced by changing the number of positions in the secret code and the number of colors to choose from. These adaptations influence the difficulty of the reasoning about the codes, which is part of the first task. However, as explained in section 4.1, using fewer positions and colors has some disadvantages.

An improvement in the experimental setup should be made to better be able to measure complex skills and strategies. As has been mentioned before, participants with pragmatic language use had a disadvantage in strategy development. A strong strategy for this game is to reveal little information, and participants with pragmatic language use had less means of doing so than participants with a logical language use. By including more expressions, such as for example *niet alle* (*not all*), the possibilities for pragmatic language users can be increased. The pragmatic meaning of 'Not all colors are right.' would allow *exactly two or exactly three* colors to be right, which is less informative than for example the pragmatic meaning of 'Some colors are right.', which is *exactly two colors are right*. Thus, by including *niet alle*, pragmatic language users would be able to be less informative without lying. Still, more informative expressions would be preferred according to Grice's quantity maxim. Expressions from other scales could also be included, for example scalar terms meaning *more than*, *over*, *at most*.

It might also be useful to increase the number of times that questions are asked, such that the possible development of strategy and theory of mind use can be measured more closely. The risk of this is boring participants by repeatedly asking similar questions.

During the experiment, some participants got tired. Fatigue could be measured by determining physical measures, e.g. heart rate and blood pressure. This way, it could be measured to what extent advanced cognitive skills suffer from fatigue, which could be a measure for how much effort they require and thus how well they are mastered.

Hypothesis 2

A weaker alternative for hypothesis 2 could be: *In an uncooperative conversation, some people will show less pragmatic language use (Not fully in accordance with Grice's quantity maxim)*. To test this hypothesis, it should be investigated whether the cooperativeness of the situation has an influence on language use. This could be done by observing the language use of the participants who had a logical language use during the Master(s)Mind(s)-experiment, while they play a fully cooperative game, in which a mutual goal has to be reached by two or more players.

If there is a difference in how pragmatic their language use is, this is some evidence that the cooperativeness of the situation has an influence. If there is no difference, this is evidence that the cooperativeness of the situation does not have an influence. However, language use may also be strongly dependent on the opponent/ partner and the knowledge both players have about each other. Ideally, participants should not know each other beforehand, so that they do not know anything about the language use of the other player.

Apart from cooperativeness of the conversation, the influence of other aspects on language use should be tested such as: the order of the ToM reasoning used by participants, the experience participants have in the use of logics, participant's sensitivity to social aspects. There have already been studies investigating the relation between age and language use, for example [30].

Hypothesis 3

Although based on the Master(s)Mind(s)-experiment hypothesis 3 should be rejected, Feeney et al. have found evidence that the automated process which results in pragmatic interpretation, can be overruled by a deliberate reasoning process, resulting in logical interpretation for the English quantifier 'some'.

The people in the Master(s)Mind(s)-experiment who shifted to more pragmatic language use, only shifted the meaning of 'sommige' *some* from *exactly two or exactly three* to *exactly two* or the meaning of 'de meeste' *most* from *exactly three or exactly four* to *exactly three*. The fact that some people shifted to a more pragmatic language use is the reason that from the Master(s)Mind(s)-experiment, pragmatic language use cannot be said to be an automated process. As described earlier, this might be because the pragmatic meaning of quantifiers depends on the situation.

There are two ways to escape the conclusion that pragmatic language use is not fully automated in the context of the Master(s)Mind(s)-experiment. One is to state that all participants who did not start with the interpretation that 'sommige' can only mean *exactly two* were already overruling their pragmatic interpretation, and that some participants stopped doing so during the game. The other is that in this version of Master(s)Mind(s), the pragmatic meaning of 'sommige' is not strictly bound to *exactly two*, but that it can be *exactly two or exactly three* as well.

Feeney et al. found no difference in reaction times between pragmatic and logical responses for participants who gave both pragmatic and logical responses. This is not quite in line with the theory that pragmatic responses have to be overruled. It would mean that for some reason, these participants did not have to overrule their pragmatic interpretation. The group of people that gave both pragmatic and logical responses was quite large, 43%. There may be another reason for the participants who gave logical responses only, to have longer reaction times.

To make it more clear whether or not logical language use can only result from overruling pragmatic language use, it would be interesting to let the participants to the Master(s)Mind(s) experiment do an experiment like the one Feeney et al. have conducted. This could also be done for other scalar terms than *some*. Such an experiment could reveal whether the participants who had a logical language use from the start still need to overrule their pragmatic language use. If participants were to complete such an experiment before and after doing the Master(s)Mind(s)-experiment, it could also be measured whether reaction times decrease for people who have shifted to a more pragmatic use. If so, this would indeed indicate automation. On the other hand, people who have shifted to more logical use are expected to have increased reaction times, since they now have to overrule their automated interpretation process.

Cognitive Modeling

In addition to conducting more experiments, cognitive modeling could also be used to find answers to the remaining questions. This could be particularly helpful in determining what kind of reasoning processes, automated or deliberate, are involved in using scalar terms and theory of mind reasoning. Also, it could be investigated what parameters, such as for example working memory capacity, correlate with the use of a particular order of ToM reasoning and a particular type of language use.

Modeling in ACT-R is taken as an example to explain this, for two reasons. The first is ACT-R's suitability for the task at hand. ACT-R is a cognitive architecture in which a lot of attention has been paid to cognition and learning mechanisms, which links up with the subject of cognitive skill acquisition very well. At the same time, simple perceptual and motor actions required for the described experiments can be modeled in ACT-R. There are also disadvantages of using ACT-R, ACT-R does not have an advanced language module. Also, to make a Lisp model of playing Master(s)Mind(s), which is required for cognitive modeling in ACT-R, some simplifications have to be made. Because the applications of two participants interact through simple text files, it should not be too difficult to model an environment without the graphical details. This will be enough to study advanced cognition, but it will not enable detailed modeling of the perceptual-motor part of the task, nor to determine how the interface of the computer program affects performance. Creating a graphical version of Master(s)Mind(s) in Lisp is not straightforward. The second reason for choosing ACT-R as an example is the author's familiarity with this particular architecture.

An elaborate explanation of ACT-R and the ideas behind it can be found in [1]. ACT-R is an architecture of cognition and can be thought of as a framework for modeling all kinds of tasks involving cognition, that humans can do. ACT-R consists of different modules, which correspond for example to the visual system, the hands, memory. In addition, there are certain learning mechanisms on the symbolic, as well as on the subsymbolic level. ACT-R is not embodied, it operates in a simulated environment.

Automated reasoning processes are associated with procedural memory. In ACT-R, procedural memory is associated with the production rules. A production rule consists of a condition part, which should be matched for the production rule to be able to 'fire', and an action part, which is executed if the production rule fires. ACT-R has a mechanism which selects production rules based on their expected utility, if more than one production rule's conditions are met. If pragmatic language use results from automated processes, it should be possible to model this type of language use in ACT-R, using mainly the production rule system.

Deliberate reasoning processes on the other hand, are associated with declarative memory. Declarative memory corresponds to the declarative module in

ACT-R, which contains chunks in which knowledge is stored. The declarative module interacts with the production rules through the retrieval buffer, in which a chunk can be stored such that it can be used by production rules. Production rules can request a chunk with a particular feature to be placed in the retrieval buffer. Whether or not such a request succeeds, depends on the activation of a chunk. Activation decays with time and is increased when the knowledge that the chunk represents is 'used', for example when a chunk is used by a production rule or when an object that the chunk represents is seen.

Another module of ACT-R that can be associated with deliberate reasoning processes is the intentional module. This module interacts with the production rules through the goal buffer. The goal buffer contains the current goal. Depending on what the goal is, certain productions will or will not be able to fire.

If deliberate reasoning processes are necessary for reaching expert level performance in playing Master(s)Mind(s), this should be reflected in the ACT-R model of this task, by extensive use of the intentional or declarative module.

ACT-R predicts activity in certain brain regions, when certain modules are used. The production rules are associated with activation in the basal ganglia. Use of declarative memory is associated with activation in the temporal cortex and hippocampus. The brain region for the intentional module has not yet been identified. The retrieval and goal buffer correspond respectively to the ventrolateral and the dorsolateral prefrontal cortex. This way, an ACT-R model leads to predictions which can be tested using fMRI scans.

ACT-R is also capable of modeling individual differences. This can be done in several ways. One is to vary the value of certain parameters. The parameter that is associated with working memory capacity is W. There are also parameters which influence how easily chunks are retrieved, how fast activation of chunks decays, how much activation is increased, etc. It could be tried to model the individual differences found in the Master(s)Mind(s)-experiment, by varying parameters.

Another way to model individual differences is to implement different strategies by writing different production rules. A new strategy can also be learned by the model, because there is a learning mechanism, called 'production compilation', which can combine two production rules into one new rule under certain circumstances. If a new production rule, that is created by production compilation, is more efficient than the old rules, it will eventually get a higher utility, based on sub symbolic learning, and will therefore eventually be preferred most of the time. However, this will not result in deliberate reasoning processes. Production-compilation transfers knowledge from the declarative to the procedural memory, and corresponds to automation rather than reflection. It could be the case that a new learning mechanism is needed to model reflection. Cognitive modeling and cognitive psychology could both contribute to a theory about such a mechanism.

In [27], Misker and Anderson combined ACT-R and Optimality Theory. There are no models of theory of mind reasoning in ACT-R yet.

Chapter 7

Application of Theories

In this chapter, it is shown how modal epistemic logic can be used to describe the reasoning that participants used during the experiment, and how modal epistemic logic can help to determine the order of ToM used. In addition, an explanation is given of how weak bidirectional optimality theory can be applied to the situation of playing Master(s)Mind(s).

7.1 Modal Epistemic Logic

During the experiment, participant 1 wrote in an answer to a question that in making a guess, he considered that the guesses made by his opponent were evasive from his opponent's secret code, since he wanted to hide this code. Participant 1 was applying the knowledge that his opponent wanted to hide his secret code, in other words, that his opponent did not want him to know his secret code. Let c denote *the secret code of participant 1's opponent*, and let participant 1 and his opponent be denoted by agent 1 and agent 2 respectively. If the operator D_i is used for desire, just like K_i is used for knowledge, the fact that participant 1 has this knowledge could be represented as follows in modal epistemic Logic: $\mathrm{K}_1\mathrm{D}_2\neg\mathrm{K}_1 c$.

As explained in section 2.2, the order of this knowledge can be found by counting the operators, as long as the agents are different and the first operator, which serves to indicate which agent has the knowledge, is left out. The representation of the knowledge participant 1 has (as opposed to the representation of the fact that participant 1 has this knowledge) is: $\mathrm{D}_2\neg\mathrm{K}_1 c$. Now there are two operators with different agents, of which the first is not the agent that has the knowledge and thus it can be concluded that this knowledge is part of a second order ToM. This can also be derived as follows: c is zeroth order knowledge and is attributed to agent 1 by agent 2, thus agent 2 is attributed a first order ToM. The knowledge of agent 1 that agent 2 has a first order ToM is second order knowledge.

Participant 1 also wrote that he tried to make guesses which did not resemble his own code too much nor too little. Apparently, participant 1 applied the knowledge that he did not want his opponent to know his secret code. If c now denotes *participant 1's secret code*, this desire can be represented in modal epistemic logic as follows: $\mathrm{D}_1\neg\mathrm{K}_2 c$. Leaving out the first operator, and then counting the operators left, it is found that this desire is part of a first order ToM. This is indeed correct, because agent 1 is attributing zeroth order knowledge to agent 2.

Modal epistemic logic provides a way to formalize the knowledge, desires and intentions of the participants to the Master(s)Mind(s)-experiment. In addition, it provides an easy way to determine what order of ToM is required to have certain knowledge, desires or intentions.

7.2 Bidirectional Optimality Theory

Stages in development and in OT

Experiments by Noveck [29] and Papafragou and Musolino [30] have shown that there are at least two stages in interpreting scalar terms. In the first stage, children show a logical way of interpreting scalar terms. They do not draw pragmatic inferences, like adults do. In the second stage, adults are sensitive to pragmatic implicatures and have a pragmatic way of interpreting scalar terms. Feeney et al. [11] have presented some evidence for a third stage, in which language use can be logical again.

In Optimality Theory, the first stage of development corresponds to using unidirectional OT as optimization process. Unidirectional OT thus results in logical language use. Once children have learned to use bidirectional OT as optimization strategy, language use is pragmatic. In bidirectional OT, interpretations are strengthened as a result of the application of knowledge about the speaker's alternatives and knowledge. Because superoptimal forms get a meaning as well, the meaning of the optimal forms becomes more specific. This way, the form-meaning pairs that pragmatic language use consists of, become (super)optimal. Pragmatic interpretations are a subset of truth-conditional interpretations. It is not yet clear what optimization process is used in the third stage of development, that is to say if a third stage exists.

In the Master(s)Mind(s)-experiment, some participants started with or developed pragmatic language use. Only one participant was very logical in his language use. A lot of participants had a language use that was neither fully logical nor fully pragmatic. It is not plausible that some of the participants were not able to use bidirectional OT as optimization process, because they were all adults. The participants that did not have a pragmatic language use may have used a different optimization process than uni- or bidirectional OT. Another way in which OT can explain individual differences is by assuming a difference in constraint ranking. It could also be the case that some participants were using marked forms with marked meanings, in other words, chose to use superoptimal form-meaning pairs. However, this last option has a problem.

In playing Master(s)Mind(s), logical and pragmatic language users would sometimes choose the same form. Let us consider the quantifier scale. If the current state of the world is that exactly one color is right, the best form to use according to a participant with pragmatic language use is 'een'. This is the form a participant with logical language use would choose in this situation as well. In fact, the logical language user would choose 'een' in all cases except when no colors are right, because to him 'een' means *at least one*. If the logical optimum and the pragmatic optimum have the same form, but a different meaning, the logical optimum could never be superoptimal if the pragmatic optimum is optimal, since in the second round, all candidates with the same form are excluded. Thus, weak bidirectional OT by itself cannot account for participants using the same form, but with a different meaning.

An OT analysis of language use in Master(s)Mind(s)

Since the interest of this study is in the transition from the second stage of development to a third stage, weak bidirectional OT is used as the optimization process in the analysis below. One constraint is formulated to model the situation of playing Master(s)Mind(s), and the resulting optimal form-meaning pairs are presented.

In playing Master(s)Mind(s) there are five possible worlds: the world where no colors are right, the world where exactly one color is right, etc. (The possibility of no colors being right is included. In the experiment this was not possible, since four different colors had to be chosen out of seven colors. Including this possibility could correspond to the situation where there were eight colors to choose from.) Meanings specify what worlds can be possible, for example '1' for the world where exactly one color is right to be the possible, and '2' for the world where exactly two colors are right to be possible.

The constraint needed is a general constraint, which demands that forms have a meaning which is consistent with their truth-conditional meaning:

Truth-Conditional Meaning A form must be used with a meaning that is (a subset of) the truth-conditional meaning of that form.

This constraint is violated whenever a form is used with a meaning that is not (a subset of) its truth-conditional meaning. The effect is that form-meaning pairs such as (een, 0), and (alle, 2) are excluded. Table 7.1 lists the truth-conditional meanings of the quantifiers, for the domain of playing Master(s)Mind(s). Meanings for the numbers are similar, thus 'één' means at least one, etc.

Table 7.1: Truth-conditional meaning for the quantifiers in the domain of playing Master(s)Mind(s).

Quantifier:	Truth-conditional meaning:
geen	none, thus 0
een	at least 1, thus 1 or 2 or 3 or 4
sommige	at least 2, thus 2 or 3 or 4
de meeste	more than half, thus 3 or 4
alle	all, thus 4

To illustrate the way in which weak bidirectional OT can be applied to the situation of playing Master(s)Mind(s), a diagram (table 7.2) is used instead of an OT tableau. This type of diagram was introduced by Blutner in [5], page 25. The first column shows the forms from which participants could choose, the first row shows the possible meanings for these forms. Each cell below the first row and left to the first column represents a form-meaning pair.

An 'F' is used to indicate that a form-meaning pair violates the constraint Truth-Conditional Meaning. This corresponds to the grey area in Blutner's diagram. After this constraint has been applied, it can be seen that (geen, 0), (alle, 4) and, (een, 1) are optimal form-meaning pairs. This is because they are the only candidates for 'geen' and '0', 'alle', and '1', respectively. The optimal form meaning pairs are indicated in the diagram with 'I'.

For the second round, all candidates with the same form or the same meaning as one of the optimal form-meaning pairs are excluded, because they are blocked by the optimal pairs. 'B' is used in the diagram to indicate which form-meaning pairs are blocked. 'II' is used to indicate the superoptimal form-meaning pairs.

In the second round, (sommige, 2) is superoptimal, because it is the only candidate left for the meaning '2'. (de meeste, 3) is superoptimal as well, because this is the only candidate for the form 'de meeste'. This causes (sommige, 3) to be blocked after the second round, which is indicated in the diagram by B_2.

Table 7.2: Diagram which illustrates the application of weak bidirectional OT to the game Master(s)Mind(s) for the quantifier scale. Forms are listed vertically and meanings are listed horizontally. The cells indicate form-meaning pairs. 'F' means that a form-meaning pair violates the constraint Truth-Conditional Meaning. 'B' means 'blocked'. 'I' means optimal and 'II' means superoptimal.

	0	1	2	3	4
geen	I	F	F	F	F
een	F	I	B	B	B
sommige	F	F	II	B_2	B
de meeste	F	F	F	II	B
alle	F	F	F	F	I

The diagram in table 7.3 is similar to the one in table 7.2. This diagram shows the results for the number scale in the game Master(s)Mind(s).

Table 7.3: Diagram which illustrates the application of weak bidirectional OT to the game Master(s)Mind(s) for the number scale. Forms are listed vertically and meanings are listed horizontally. The cells indicate form-meaning pairs. 'F' means that a form-meaning pair violates the constraint Truth-Conditional Meaning. 'B' means 'blocked'. 'I' means optimal and 'II' means superoptimal.

	0	1	2	3	4
één	F	I	B	B	B
twee	F	F	II	B_2	B
drie	F	F	F	II	B
vier	F	F	F	F	I

In the experiment, the two scales (quantifiers and numbers) were in competition with each other. This competition is not yet modeled in OT. Participants did not consistently prefer one scale above the other. The choice of scale may depend on factors that are not linguistic.

It can be seen from table 7.2 and 7.3 that the optimal and superoptimal form-meaning pairs are the form-meaning pairs that pragmatic language use consists of. Which of these form-meaning pairs was chosen by a participant depended on the situation (e.g., how many colors were correct) and the game rule that lying was not allowed. The meaning of the form used had to contain the current state of the world.

Although not necessary for the above analysis, some additional constraints may be useful in future work. Some ideas considering such additional constraints are presented below.

It is not straightforward to capture Grice's quantity maxim in OT. Remember that constraints can only prohibit or demand something, they should not compare different candidates, since this should be done by Eval. Comparing different candidates' informativeness is exactly the way an optimum is found according to Grice's quantity maxim. The quantity maxim states that speakers should be as informative as possible. For the quantifier scale in the domain of playing Master(s)Mind(s), this can be captured in the following constraints:

Not 'een' Do not use 'een'.

Not 'sommige' Do not use 'sommige'.

Not 'de meeste' Do not use 'de meeste'.

If these constraints are ordered in the way they are presented above, they have the effect that more informative terms are preferred above less informative terms. They would result in vertical arrows in the diagram from one cell to a cell that represents a (super)optimal form-meaning pair, indicating preferences. An explanation of the meaning of these arrows can be found in [5]. These constraints are inspired by the quantitative measure for relevance that Van Rooy provides in [36]. This measure can be seen as a measure for informativeness. In the context of the Master(s)Mind(s)-experiment, this measure results in the scale ⟨geen/ alle, de meeste, sommige, een⟩, which is ordered from most informative to least informative.

The following constraint is from the perspective of the hearer:

Unique meaning A form has a meaning which includes only one possible world.

This constraint corresponds to horizontal arrows in the diagram. Once a form-meaning pair, or several form-meaning pairs are (super)optimal, this constraint is violated by all form-meaning pairs that have the same form and a different meaning. This has the effect that, for example, the meaning *exactly one* will be preferred over the meaning *exactly one or exactly two*.

Comparison of the analysis and the data

The analysis presented above, cannot explain the logical language use that participants displayed during the experiment. However, it can explain the pragmatic language use of some participants and the shift to more pragmatic language use that some participants made.

From table 5.1 on page 38, it can be seen that participants 6, 7, 9, and 12 first used 'sommige' with the meaning *exactly two or exactly three* and later on with the meaning *exactly two*. This could be because the form-meaning pair (sommige, 3) is blocked only after the second round of weak bidirectional OT, and these participants needed some practice to use this blocking effect in their interpretation. When looking at the quantifier 'de meeste' it can be seen that participants 6, 7, and 9 had the interpretation that 'de meeste' means *exactly three* from the start (participant 12 did not use 'de meeste'). This interpretation is found in the second round and does not depend on blocking effects which occur after the second round. The same holds for the pragmatic interpretation of the other quantifiers and these participants used these pragmatic interpretations from the start.

Given these results, it may be the case that weak bidirectional optimization up to two rounds is fully automated in the context using the quantifier scale while playing Master(s)Mind(s), whereas higher rounds of weak bidirectional optimization are not. For the number scale, similar effects were not found. As described before, the use of this scale is less context dependent and therefore pragmatic use of numbers may be fully automated in the context of playing Master(s)Mind(s).

If the shift to more pragmatic language use indeed occurs because participants still have to learn to use the blocking effects that occur after the second round of weak bidirectional optimization, this is some evidence for hypothesis 1, which stated that playing the game Master(s)Mind(s) and reflecting upon this, is a form of dual tasking. An explanation for the shift could be that only after the game is mastered well enough, there is enough capacity left to use higher rounds of weak bidirectional optimization.

The presented OT analysis cannot explain logical language use in a possible third stage. Logical language use in the first stage can be explained as being the result of a different optimization strategy. When using unidirectional optimization, no blocking effects occur and all the form meaning-pairs that do not violate Truth-Conditional-Meaning are optimal. This results in logical language use.

Chapter 8

Conclusion

To investigate to what extent people acquire and use complex skills and strategies in the domains of reasoning about others and language use, an experiment was conducted in which participants played Master(s)Mind(s) repeatedly, to allow for development. Twelve participants took part in the experiment, of whom nine completed two sessions.

It was found that some participants used the complex skill of second order theory of mind reasoning from the domain reasoning about others. In the domain of language use, some participants used the complex skills of drawing pragmatic inferences and others used the skill of logical language use. In addition, some people used the strategy of considering the amount of information to be revealed as a result of the guesses they made. It can thus be concluded that some participants used complex skills and strategies in the domains of reasoning about others and language use, while playing Master(s)Mind(s). There clearly were individual differences: Some participants did not seem to use complex skills and strategies.

It was not found that participants acquired complex skills and strategies while playing Master(s)Mind(s). The participants who made use of such skills and strategies already did so very soon in the experiment, when it was first measured. Some development was seen, but overall development was very limited.

Hypothesis 1 stated that performing a task and simultaneously reflecting upon this task is a form of dual-tasking. It could be the case that playing Master(s)Mind(s) can be seen as a dual-tasking situation, where the first task is to play the game according to its rules and to reason based on truth-conditional meaning, and the second task is to develop a strategy based on ToM reasoning and reasoning from implicated meaning. Although no convincing evidence was found for this hypothesis (hypothesis 1), no convincing evidence was found against it either.

Hypothesis 2 stated that in an uncooperative situation, people will shift their interpretation and production of quantifiers from pragmatic (using Grice's quantity maxim) to less pragmatic (not using Grice's quantity maxim). None of the participants developed a more logical language use in the uncooperative context of playing Master(s)Mind(s), in the way that was meant in hypothesis 2. Some participants did use logical language use though, but they did so from the start. The same holds for second order ToM reasoning. It can therefore be concluded that complex skills can be transferred from other domains to the domain of playing Master(s)Mind(s).

Hypothesis 3 stated that in interpreting and producing quantifiers, people make use of an automatic process, which results in a pragmatic use of the quantifier, and that this automated process can be 'overruled' by a deliberate reasoning process, which results in a logical use of the quantifier. It is clear that not all adults have to use pragmatic language use all of the time. They can choose to use more logical language use. The experiment does not make clear whether or not this is the result of an automated process being overruled by a deliberate reasoning process. From the experiment, it seems that pragmatic language use is not automated for all people, since some participants developed pragmatic language use while repeatedly playing Master(s)Mind(s). However, there are two ways to escape this conclusion, as described in section 6.3. One is to state that some participants were already overruling their pragmatic interpretation at the start of the experiment, and some participants stopped doing so during the experiment. The other is to state that the pragmatic meaning of 'sommige' can be either *exactly two* or *exactly two or exactly three*.

Bibliography

[1] J.R. Anderson, D. Bothell, M.D. Byrne, S. Douglass, C. Lebiere, Y. Qin. An integrated theory of the mind. *Psychological Review* 111.4: 1036-1060. American Psychological Association, 2004.

[2] J. Atlas, S. Levinson. It-clefts, informativeness and logical form. *Radical Pragmatics*: 1-61. Academic Press, New York, 1981.

[3] D. Beaver, H. Lee. Input-Output Mismatches in OT. In: R. Blutner & H. Zeevat (eds.). *Optimality Theory and Pragmatics*. Palgrave/ Macmillan: p112-153, 2004.

[4] K. Binmore. *Fun and Games, A Text on Game Theory*. D. C. Heath and Company, Lexington, MA, 1992.

[5] R. Blutner. Pragmatics and the Lexicon. In: L.R. Horn, Ward, G. *Handbook of Pragmatics*. Oxford, Blackwell, 2004.

[6] R. Blutner, T. Solstad. Two case studies in lexical pragmatics. In E. Németh, K. Bibok (eds.) *Pragmatics and the Flexibility of Word Meaning.*: 11-28. Elsevier Science, Amsterdam, 2001.

[7] R. Blutner. Some Aspects of Optimality in Natural Language Interpretation. *Journal of Semantics*: 17:189-216, 2000.

[8] H.H. Clark, C.R. Marshall. Definite reference and mutual knowledge. In H.H. Clark. *Arenas of Language Use.*: 9-52. University of Chicago Press, 1992.

[9] P. Dekker, R. van Rooy. Bi-directional Optimality Theory: An application of game theory. *Journal of Semantics*, 17: 217-242, 2000.

[10] J. St. B. T. Evans. In two minds: dual-process accounts of reasoning. *TRENDS in Cognitive Sciences*, 7(10): 454-459, 2003.

[11] A. Feeney, S. Scrafton, A. Duckworth, S.J. Handley. The story of some: Everyday pragmatic inference by children and adults. *Canadian Journal of Experimental Psychology*, 2004.

[12] L. Gazendam. *Menselijk Beslissen in Theorie en Praktijk tijdens het Herhaalde Prisoners Dilemma*. Master's Thesis, University of Groningen, 2004.

[13] P. Grice. *Studies in the Way of Words*. Cambridge, MA: Harvard University Press, 1989.

[14] J. Grimshaw, V. Samek-Lodovici. Optimal subjects and subject universals. In P. Barbosa, D. Fox, P. Hagstrom, M. McGinnis & D. Pesetsky, *Is the Best Good Enough?* Cambridge, Mass.: MIT Press, 1998.

[15] P. Hendriks, H. de Hoop. Optimality Theoretic Semantics. *Linguistics and Philosophy* 24.1: 1-32, 2001.

[16] P. Hendriks, J. Spenader. A bidirectional explanation of the pronoun interpretation problem. *Proceedings* of the ESSLLI Workshop on Semantic Approaches to Binding, Nancy, France, August 16-20, 2004.

[17] W. S. Horton, B. Keysar. When do speakers take into account common ground? *Cognition* (59): 91-117, Elsevier Science B.V., 1996.

[18] L.M. Hyman. Suffix ordering in Bantu: a morphocentric approach. *Yearbook of Morphology 2002*: 245-281, 2003.

[19] A. Karmiloff-Smith. *Beyond Modularity: A Developmental Perspective on Cognitive Science*. Cambridge, MA: MIT/Bradford Books, 1992.

[20] B. Keysar, S. Lin, D.J. Barr. Limits on theory of mind use in adults. *Cognition* (89): 25-41, Elsevier Science B.V., 2003.

[21] B. Keysar, D.J. Barr, J.A. Balin, J.S. Brauner. Taking perspective in conversation: The role of mutual knowledge in comprehension. *Psychological Science*,11 (1), American Psychological Society, 2000.

[22] B. Keysar, D. J. Barr, A. Balin, T.S. Peak. Definite reference and mutual knowledge: Process models of common ground in comprehension. *Journal of Memory and Language* (39): 1-20, Academic Presss, 1998.

[23] B. Kooi. Master(s)mind(s). Unpublished manuscript. See http://www.philos.rug.nl/~barteld/.

[24] S. Kripke. Semantic analysis of modal logic. *Zeitschrift für Mathematische Logik und Grundlagen der Mathematik 9*: 67-96, 1963.

[25] J.J. McCarthy. *A Thematic Guide to Optimality Theory*. Cambridge University Press, 2002.

[26] L.W. McGarrity. *Constraints on Patterns of Primary and Secondary stress*. Unpublished doctoral dissertation, Indiana University, Bloomington, 2003.

[27] J.M.V. Misker, J.R. Anderson. Combining optimality theory and a cognitive architecture. In *Proceedings of the Fifth International Conference on Cognitive Modeling*, 165-170, Bamberg, Germany, 2003.

[28] Y. Morimoto. Antisymmetry and optimality: positional variation in negation. Paper presented at Societas Linguistica Europea, Universitt Potsdam, July 22-25, 2002.

[29] I. Noveck. When children are more logical than adults: experimental investigations of scalar implicature. *Cognition* (78): 165-188, Elsevier Science B.V., 2001.

[30] A. Papafragou, J. Musolino. Scalar implicatures: experiments at the semantics-pragmatics interface. *Cognition*(86): 253-282, Elsevier Science B.V., 2003.

[31] W. Schneider, J.M. Chein. Controlled & automatic processing: behavior, theory, and biological mechanisms. *Cognitive Science* 27: 525-559, Cognitive Science Society, 2003.

[32] W. Schneider, R.M. Shiffrin. Controlled and automatic human information processing. I. Detection, search, and attention. *Psychological Review*, 84(1): 1-66, American Psychological Association, 1977.

[33] H. Simon. *Models of Man*. New York: Wiley, 1957.

[34] R. Sun, X. Zhang. Top-down versus bottom-up learning in cognitive skill acquisition. *Cognitive Systems Research*, 5: 63-89, Elsevier B.V., 2004.

[35] W. van der Hoek, J.-J. Ch. Meyer. *Epistemic Logic for AI and Computer Science* (Cambridge Tracts in Theoretical Computer Science, No 41), Cambridge University Press, 1995.

[36] R. van Rooy. Relevance and Bidirectional OT. In R. Blutner and H. Zeevat(eds.). *Pragmatics in Optimality Theory*. Palgrave/ Macmillan, p173-210, 2003.

[37] R. Verbrugge, P. Hendriks, N. Taatgen et al. Bidirectionality in Skill Acquisition: From learning by automation to learning by reflection (and back). Research proposal, 2004.

Appendix A

Master(s)Mind(s) Implementation

In this appendix the general idea and some of the details of the implementation of Master(s)Mind(s) in Kylix are explained. The first part explains how it was achieved that players could play the game head to head via a network. The second part contains a description of the screens that were used. The third part contains two examples of how screens were updated while playing the game. In the first of these examples, the way the two applications communicated is illustrated as well.

Design

The implementation of Master(s)Mind(s) enabled players to play against each other. This was achieved by having the code for player 1 and the code for player 2 in the same project. The resulting application was then run on two connected computers, one for each player.

A boolean was used to register whether the application was used for player 1, which was the player who could make a guess in the odd rounds, or for player 2, who made a guess in the even rounds. Dependent on the value of this boolean, different parts of if-loops were executed.

The two running applications communicated with each other through text files. One file was used to which player 1 wrote data, and from which player 2 read data, and one file was used to which player 2 wrote data and from which player 1 read data. In addition, there was a third file to which both players wrote data. In this file, the relevant data of the experiment were collected.

To synchronize the two applications, not only data considering the game, but also a number, which corresponded to what phase of the game a player was in, was written to the player's file. For example, after player 1 had made a guess, his phase was increased from 1 to 2. This way, the application of player 2 could read from player 1's file whether or not player 1 was already done guessing.

Screens

The application consisted of fourteen screens in total. The first screen that was shown was the main screen, in which the experimenter could submit relevant data such as the name of the player, how often the player had played the game before, whether questions should be asked during the game, whether the player should start guessing, and what files to use.

The actual game started with a screen that enabled the players to choose a secret code. In this screen a palette with seven colors was shown to choose from, which consisted of a panel with shapes, and an instruction was given on how to choose a secret code. If the player chose a code that was not allowed, there was a screen to point this out and to give an extra instruction.

Then the most important screen, the overview screen was shown (see figure A.1 on page 72). This screen provided three overviews. Leftmost was an overview of codes guessed by the player, and the feedback received and given about these codes. The overview was a panel, with panels for each turn in which the player could guess. In the middle of these smaller panels were four shapes (circles) to represent a code. On the left and on the right were labels, on which the feedback sentences were represented. Under this overview was a color palette, which the player could use to guess a code. Colors from this palette could be dragged to shapes in the overview. A lighter color was used to indicate the present turn.

In the middle there was a similar overview of the codes guessed by the player's opponent. Underneath this overview was the secret code the player had chosen. The rightmost overview was an overview of available feedback sentences. This overview consisted of a listbox. When enabled, the overview of feedback sentences could be used to select and submit feedback sentences. If the player had chosen incomplete feedback, there was a screen shown which pointed this out and gave an extra instruction. In the lower right corner was an instruction window, which provided an instruction on what the player should do next.

A tab-sheet with hidden tabs was used such that players could also evaluate a code (sheet 2), and give their interpretation of feedback sentences (sheet 3) in the overview screen. Dependent on the phase of the game, different sheets were shown and hidden. The rightmost overview with feedback sentences and the instruction window remained visible all of the time.

On the evaluation sheet (see figure A.2 on page 73), the code guessed and the players own secret code were shown. Underneath was a square with four small circles, consisting of a groupbox with four shapes, that the player could use to submit his evaluation. This could be done by dragging colors from a palette below. The screen also had an instruction on how to evaluate the guessed code. If the evaluation given was incorrect, a screen was shown that explained what the correct evaluation was.

On the interpretation sheet (see figure A.3 on page 74), the feedback sentences a player received were shown on top. Below was an instruction on how to submit the interpretation. This could be done by dragging colors to a square with four circles, similar to in the evaluation screen. By submitting all of the evaluations the player thought to be possible, the interpretation could be submitted. Next to the part where evaluations could be submitted, was an overview of the evaluations submitted so far. This overview consisted of a panel with twelve (hidden) groupboxes. Only as many groupboxes as there were evaluations submitted were visible. From this overview, evaluations could be deleted via a pop-up menu. There was a special button the player could press when done, thus different from the button for submitting an evaluation. If this button was pressed while no evaluations were submitted at all, a screen was shown which asked whether the player was sure that he thought no evaluations to be possible. If a player tried to submit an evaluation that was already submitted a message was shown, as well as when a player tried to submit more than twelve evaluations.

Two screens were used to ask players questions during the game. One for a question on why the player guessed a particular code, and one on why a player gave certain feedback sentences. These screens had a large textbox in which the answer could be submitted.

Three screens were used to indicate the end of the game. One for the case that the player had won the game, one for if he had lost, and a third one for the case where the maximum number of turns had been played.

Button-presses were often used to call a procedure which took care of the proceeding of the game. In these procedures data were written to the data collection file and the file of the player, data were read from the file of the opponent and further screens were initiated.

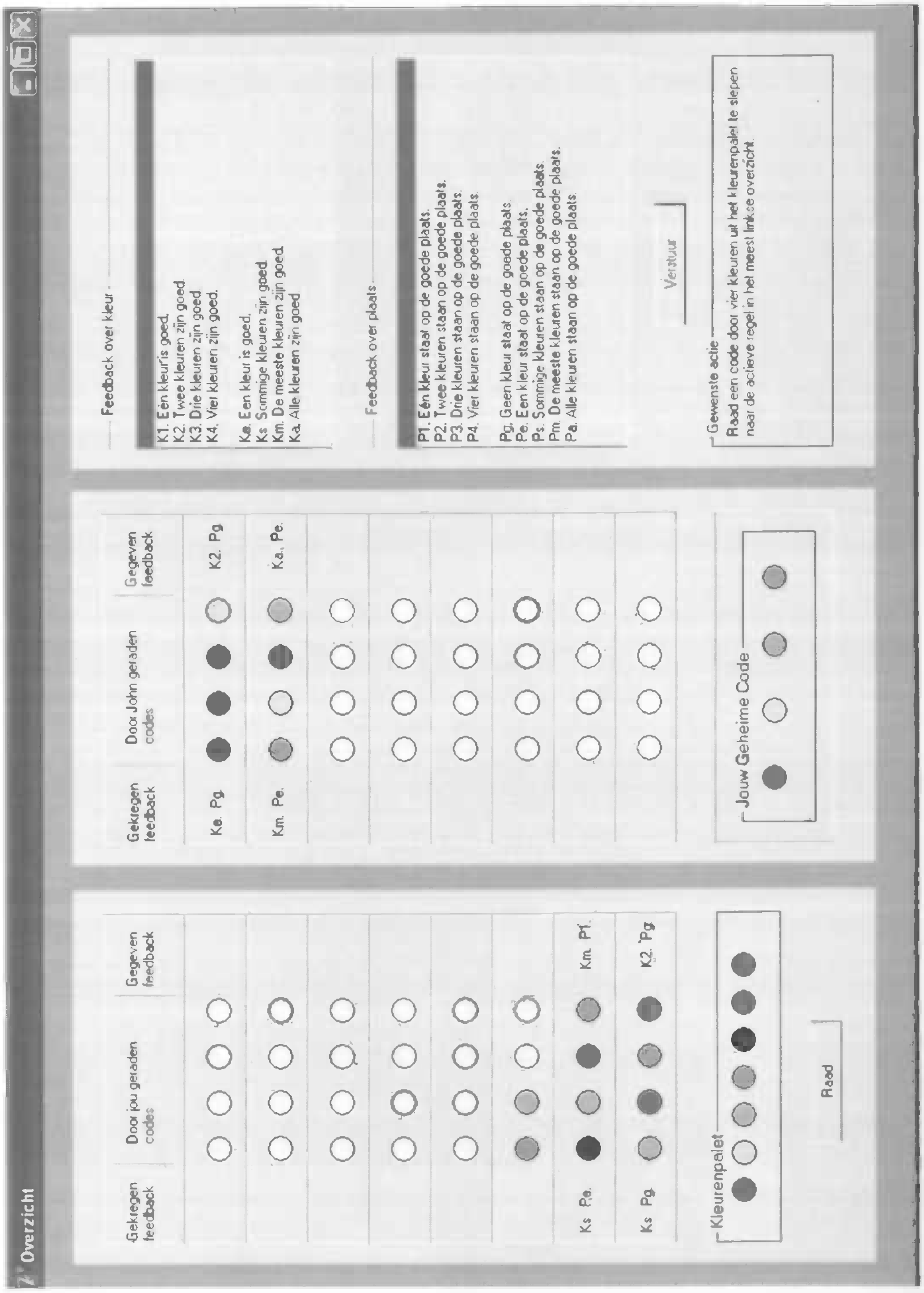


Figure A.1: Overview Screen with overview of codes

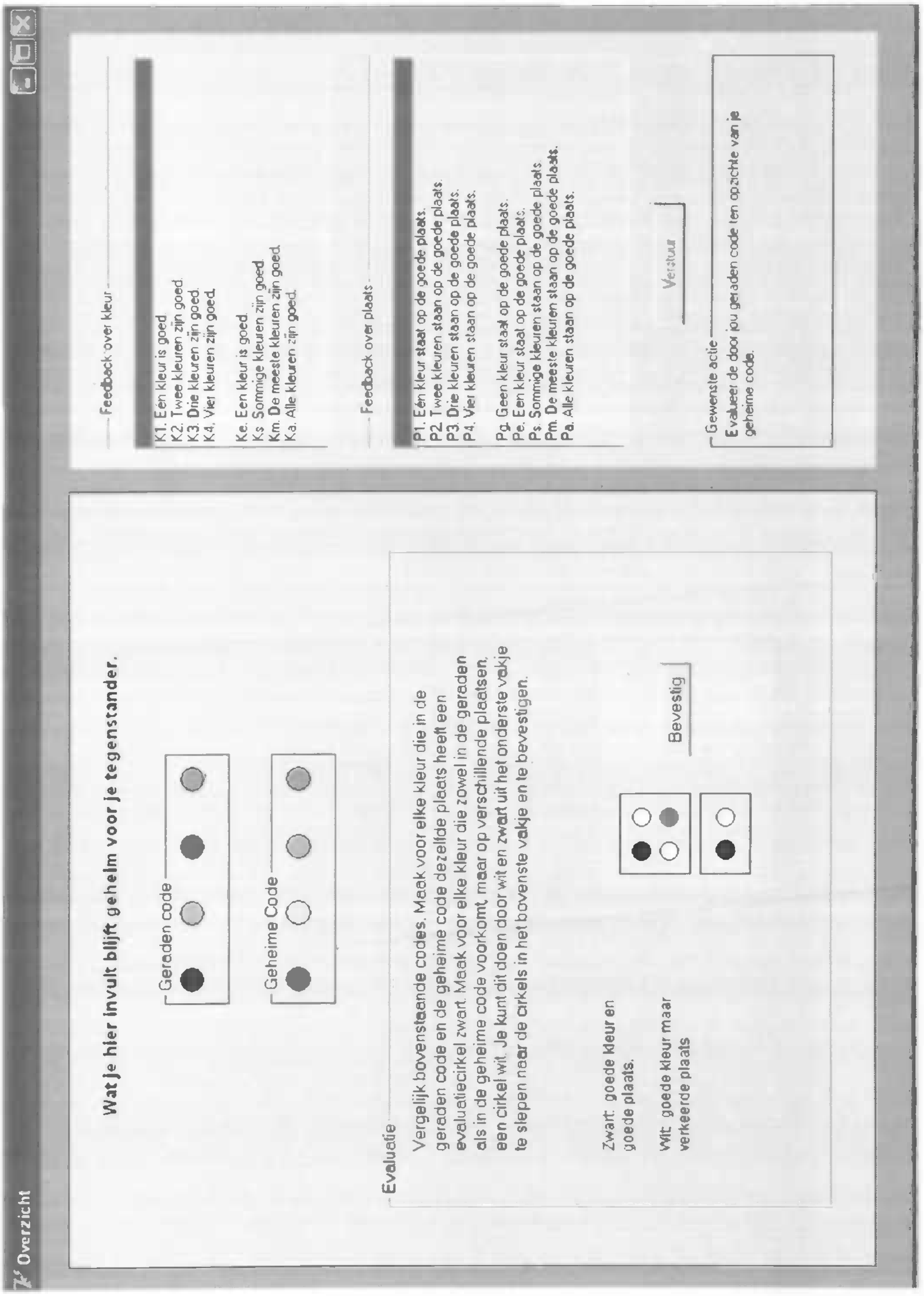


Figure A.2: Overview Screen with evaluation sheet

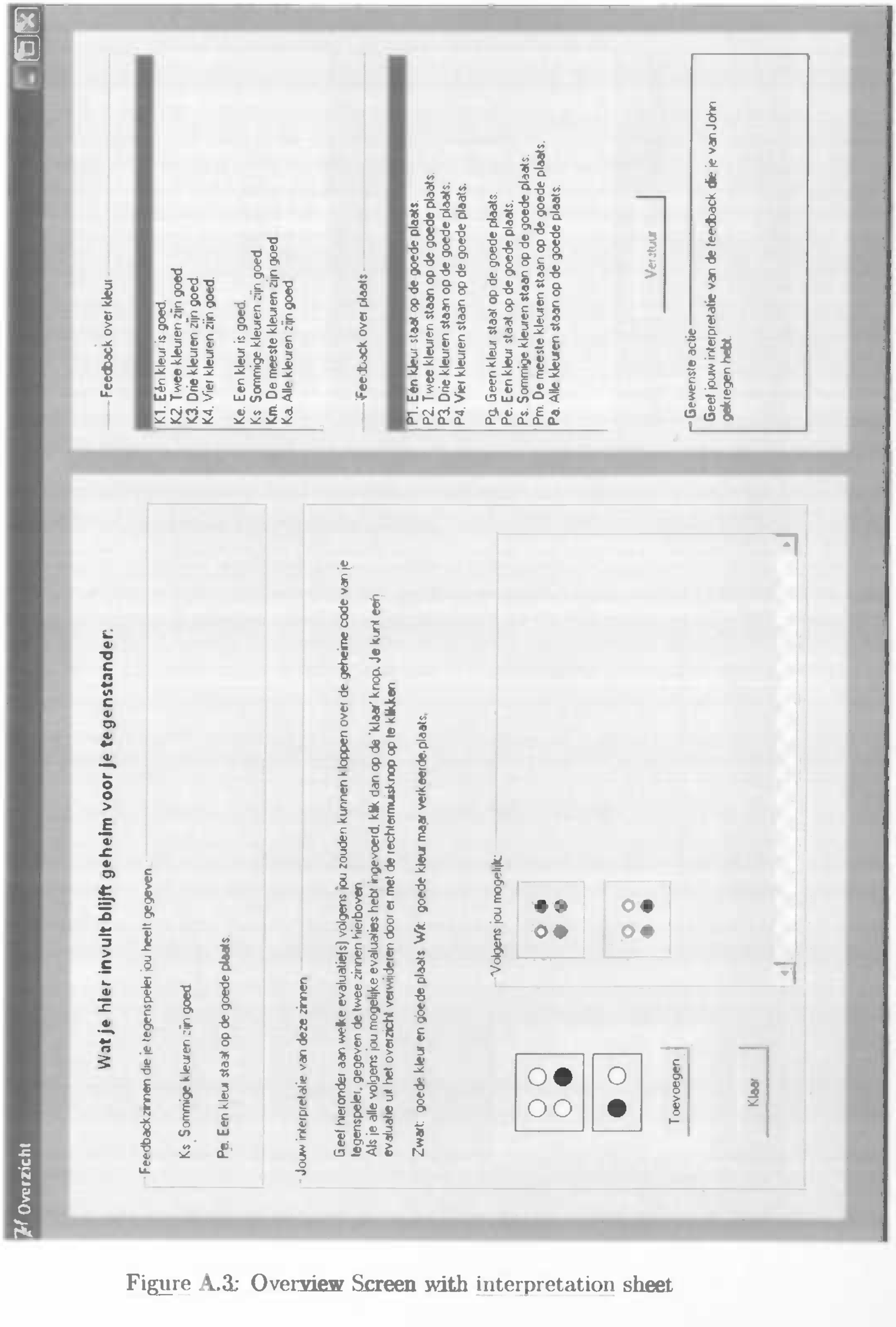


Figure A.3: Overview Screen with interpretation sheet

Updating Screens

To be able to update the overview screen of guessed codes, a two dimensional array was used, which contained all shapes that were used to represent codes. The number of the round in which the code was guessed, which was the same as the number of the panel that the shapes were on, and the place of the shape in the code were used to refer to a shape. A shape was white until it was changed into a color that corresponded to a color in a code guessed. Another two dimensional array was used which contained all labels that were used to show representations of feedback.

If player 1 guessed a code, the values of the corresponding shapes in the array were updated, as a result of dragging over colors from shapes on a color palette. This was done through events. To update the overview screen of player 2, the guessed code, which was read from the file that player 1 wrote to, was used, as can be seen in the listing below. The array geradenCode contains the guess made by player 1, shapelijst is the array with shapes.

```
if (not isSpeler1) and speler1Raadt then begin
    // lezen van door speler 1 geraden code
    repeat
      Application.ProcessMessages;
      Reset(file1);
      Readln(file1, temp);
      Readln(file1, anderFase);
      CloseFile(file1);
    until(anderFase = '2');
    geradenCode := StringToKleurlijst(temp);

    // geraden code in lijst van alle geraden codes zetten
    for i := 1 to 4 do
    begin
      shapelijst[beurtnummer, i].Color := geradenCode[i];
```

This listing also illustrates the use of the boolean isSpeler1, which was used to represent whether a player was player 1, and the boolean speler1Raadt, which represented whether it was player 1's turn to make a guess. In addition, it can be seen how different phases ('fase') were used to synchronize the two applications. After player 1 had written his guess to file1, he increased his phase from 1 to 2. The new phase was also written in file1, which enabled the application of player 2, to 'know' that player 1 had written his guess to file1. The statement Application.ProcessMessages, was used to make the screen of player 2 look good, while his application was waiting for the guess of player 1. The updating of the labels for feedback was done in a similar way.

Below, a second example is given of how some of the screens where updated while the game was being played. This example is about the screen that was used to enable a player to submit an interpretation.

In the sheet that was used to enable a player to submit his interpretation of received feedback sentences, the player was given an overview of the possible worlds/ possible evaluations submitted so far. This was done in a way similar to the overview of codes guessed, but in this overview only evaluations submitted became visible. Thus, there were no empty evaluations visible, as opposed to the empty shapes in the overview of codes.

The procedure `btnToevoegenClick`, which is partly listed below, adds submitted possible evaluations to a list of possible evaluations: `lstMogelijkheden`. This procedure is called after the player has 'pressed' the submit button. After the list has been updated, the procedure `Toonmogelijkheden` (page 77) is called, which updates the overview of submitted possible evaluations.

In `btnToevoegenClick`, the possible evaluation is first stored in a list called `evaluatie`. The properties `plaats1.Color`, `plaats2.Color` etc., correspond to colors of shapes in the screen, that the player has given a color (black or white) by dragging over colors from a color palette. This was handled by events. Then, the procedure `NieuweEvaluatie` is called, which checks if this evaluation, or an equivalent one is not already submitted by the player. If the evaluation is new, it is added to `lstMogelijkheden`.

The procedure `ToonMogelijkheden` makes use of a list of shapes, in which shapes are represented by the number of the groupbox they appear in and their position in that groupbox. A groupbox contains four shapes, which together form a possible evaluation. After the shapes have got the right color, the groupbox is made visible. All groupboxes appear on a panel which is the overview.

```
procedure TOverzicht.btnToevoegenClick(Sender: TObject);
{Voegt nieuwe evaluatie toe aan de lijst met evaluaties, laat
 mogelijke evaluaties opnieuw tonen.
}
var
  evaluatie: Tkleurlijst; //lijst van de vier ingevoerde kleuren
begin
  //inlezen ingevoerde evaluatie
  evaluatie[1] := plaats1.Color;
  evaluatie[2] := plaats2.Color;
  evaluatie[3] := plaats3.Color;
  evaluatie[4] := plaats4.Color;

  //het toevoegen van de evaluatie aan de mogelijkheden mits deze
  //nieuw is (max 12 evaluaties)
  if NieuweEvaluatie(evaluatie) then
  begin
    if Length(lstMogelijkheden) < 12 then
    begin
      SetLength(lstMogelijkheden, Length(lstMogelijkheden) + 1);
      lstMogelijkheden[Length(lstMogelijkheden) - 1] := evaluatie;
      ToonMogelijkheden;
    end else
    begin
```

```
procedure TOverzicht.ToonMogelijkheden;
{Toont de ingevoerde mogelijke evaluaties.
}
//maximaal aantal in te voeren mogelijke evaluaties
const maxNumEval = 12;
var i: integer;
begin
  for i := 0 to Min(maxNumEval, Length(lstMogelijkheden)) - 1 do
  begin
    aShapes[i,1].Color := lstMogelijkheden[i,1];
    aShapes[i,2].Color := lstMogelijkheden[i,2];
    aShapes[i,3].Color := lstMogelijkheden[i,3];
    aShapes[i,4].Color := lstMogelijkheden[i,4];

    aGroupBox[i].Visible := True;
  end;
end;
```

The procedure `mnVerwijderenClick` was used to delete a possible evaluation from the list of possibilities, after a player had selected this evaluation from the overview, to be removed. One groupbox is made invisible in this procedure, because the number of possible evaluations is decreased by one.

```
procedure TOverzicht.mnVerwijderenClick(Sender: TObject);
{Zorgt voor het verwijderen van een groupbox uit het overzicht van
 mogelijke evaluaties.
}
begin
  lstMogelijkheden := VerwijderTkleurlijst(lstMogelijkheden,
                                  (Sender as TMenuItem).tag);
  lstZwartWit := VerwijderTaantal(lstZwartWit,
                                 (Sender as TMenuItem).tag);
  //het verbergen van de laatst getoonde groupbox, omdat er een
  //verwijderd is
  aGroupBox[Length(lstMogelijkheden)].Visible := False;
  ToonMogelijkheden;
end;
```

Appendix B

Feedback sentences

B.1 In Dutch

Kleur:

1. Eén kleur is goed.
2. Twee kleuren zijn goed.
3. Drie kleuren zijn goed.
4. Vier kleuren zijn goed.

5. Een kleur is goed.
6. Sommige kleuren zijn goed.
7. De meeste kleuren zijn goed.
8. Alle kleuren zijn goed.

Plaats:

9. Eén kleur staat op de goede plaats.
10. Twee kleuren staan op de goede plaats.
11. Drie kleuren staan op de goede plaats.
12. Vier kleuren staan op de goede plaats.

13. Geen kleur staat op de goede plaats.
14. Een kleur staat op de goede plaats.
15. Sommige kleuren staan op de goede plaats.
16. De meeste kleuren staan op de goede plaats.
17. Alle kleuren staan op de goede plaats.

B.2 In English

Color:

1. One color is right.
2. Two colors are right.
3. Three colors are right.
4. Four colors are right.

5. There is a right color.
6. Some colors are right.
7. Most colors are right.
8. All colors are right.

Place:

9. One color is in the right place.
10. Two colors are in the right place.
11. Three colors are in the right place.
12. Four colors are in the right place.

13. No colors are in the right place.
14. There is a color which is in the right place.
15. Some colors are in the right place.
16. Most colors are in the right place.
17. All colors are in the right place.

Appendix C

Instruction

Het spel

doel

Je gaat zo meteen via de computer een spel spelen tegen een tegenspeler. Het doel van dit spel is de eerste te zijn, die de geheime code van zijn tegenspeler correct raadt. De code bestaat uit vier verschillende kleuren, die gekozen zijn uit zeven kleuren. Je kunt achter de code proberen te komen door te raden.

een code raden

Beide spelers kiezen een geheime code. Vervolgens mag elke beurt een speler raden naar de geheime code van de ander. Speler 1 begint hiermee. Een code die je raadt moet bestaan uit vier verschillende kleuren en je kunt kiezen uit de zeven kleuren die beschikbaar zijn voor het samenstellen van een geheime code.

evaluatie in zwart en wit geven

Nadat een code geraden is, wordt deze code door beide spelers geëvalueerd ten opzichte van hun eigen geheime code. Deze evaluatie is niet zichtbaar voor de andere speler en bestaat uit het aangeven van hoeveel kleuren er goed zijn én op de goede plaats staan en hoeveel kleuren er goed zijn, maar op de verkeerde plaats staan. Hiervoor wordt gebruikt gemaakt van de kleuren zwart en wit. Zwart geeft aan dat een kleur goed is én op de goede plaats staat, wit geeft aan dat een kleur goed is, maar niet op de goede plaats staat.

feedbackzinnen versturen

Nadat een evaluatie is ingevoerd is het tijd om feedback aan je tegenspeler te geven over de geraden code. Dit doe je als je tegenspeler een code heeft geraden en ook als je zelf een code hebt geraden. De feedback bestaat uit twee zinnen die informatie geven over hoe goed de geraden code lijkt op jouw geheime code. Als je zelf een poging doet om de code van je tegenspeler te raden, moet je dus ook iets zeggen over hoe goed deze poging is ten opzichte van jouw eigen geheime code. De feedbackzinnen kunnen worden gekozen uit een lijst. Eén zin moet gaan over kleur en één over plaats. De feedbackzinnen moeten waar zijn. Het is niet toegestaan je tegenspeler te misleiden door onware zinnen te selecteren.

interpretatie van feedback geven

Als je feedbackzinnen hebt gekregen van je tegenstander (ofwel over de door jouw geraden code, ofwel over de door hem/ haar geraden code) geef je aan hoe jij deze feedbackzinnen interpreteert. Je tegenspeler komt niets te weten over jouw interpretatie. Je kunt jouw interpretatie geven door aan te geven welke evaluaties (in termen van zwart en wit) volgens jou mogelijk zijn, gegeven de feedbackzinnen. Een evaluatie geeft dus aan hoeveel kleuren er goed zijn én op de goede plaats staan en hoeveel kleuren er goed zijn, maar op de verkeerde plaats staan. Bijvoorbeeld, je krijgt van je tegenspeler de zin:

'Niet alle kleuren staan op de goede plaats'.

Als jij vindt dat het, gegeven deze zin, niet zo kan zijn dat alle kleuren op de goede plaats staan, dan voer je geen evaluatie in met vier keer zwart. Als jij vindt dat, gegeven deze zin, het zo zou kunnen zijn dat er twee kleuren goed zijn en op de goede plaats staan, dan vul je twee keer zwart als mogelijke evaluatie in. Het is van belang dat je alle evaluaties invoert die volgens jou mogelijk zijn. Neem daar rustig de tijd voor.

Als beide spelers hun interpretatie van de feedback hebben ingevuld is de beurt ten einde en mag de speler die net niet heeft geraden een poging doen de code van de andere speler te raden.

spelverloop

Het verloop van het spel is dus als volgt:

1. Beide spelers kiezen een geheime code.

   Ronde 1:

2. Speler 1 doet een poging de geheime code van speler 2 correct te raden.
3. Beide spelers evalueren de door 1 geraden code ten opzichte van hun geheime code.
4. Beide spelers sturen feedback naar de andere speler.
5. Beide spelers geven hun interpretatie van de gekregen feedback.

   Ronde 2:

6. Speler 2 doet een poging de geheime code van speler 1 correct te raden.
7. etc.

einde van het spel

Als na ieder acht keer raden geen van de spelers erin geslaagd is de geheime code van de ander correct te raden is het spel ten einde en is er geen winnaar. Als een van de spelers de geheime code van de ander correct raadt is het spel ten einde en is degene die de code correct heeft geraden de winnaar.

Het computerprogramma

Om dit spel te kunnen spelen maak je gebruik van een computerprogramma. Het programma geeft aan wat je moet doen en hoe je dat kunt doen. Als je iets niet meteen goed doet geeft het programma dat aan. Er verschijnt dan een extra schermpje met informatie, dat je kunt sluiten. Daarna kun je het dan nog eens proberen.

Soms kan het lijken alsof het programma niet reageert. Dit is zo omdat je soms even op je tegenspeler moet wachten. Het heeft dan geen zin om herhaaldelijk op knoppen te drukken. Er verschijnt vanzelf een nieuwe instructie of een nieuw venster als je tegenspeler weer even ver is als jij.

Nadat je een geheime code hebt gekozen zie je het overzichtsscherm. Hier vind je drie overzichten.

links een overzicht van de door jou geraden codes, de feedback die je daarover gegeven hebt en de feedback die je daarover gekregen hebt,

in het midden een overzicht van de door jouw tegenspeler geraden codes, de feedback die je daarover gegeven hebt en de feedback die je daarover gekregen hebt,

rechts het overzicht van feedbackzinnen waaruit je kunt kiezen.

Rechtsonder is een vakje waarin staat wat je op dat moment moet doen. In het linker of in het middelste overzicht geeft een lichter gekleurd balkje aan welke code is geraden in de huidige beurt. Hieraan kun je zien welke code je moet gebruiken voor het evalueren en feedback geven.

De instructies zijn elke ronde waarin jij raadt en elke ronde waarin jij niet raadt hetzelfde. Het is dus niet nodig om steeds alles door te lezen als je nog weet hoe het werkt. Het programma wijst zich verder redelijk vanzelf. Als je vragen hebt over hoe het programma werkt kun je die tussentijds aan Lisette stellen.

Appendix D

Questionnaires

Vragenlijst 1

Deze vragenlijst is om in te vullen direct na het eerste dagdeel waarin je het spel gespeeld hebt.

Naam:
(die je tijdens het spel gebruikt hebt)

1. Waar let je op bij het raden van een code?

2. Waar let je op als je feedbackzinnen uitkiest?

3. Vind je dat je tegenstander dit spel slim speelt, waarom (niet)?

Vragenlijst 3

Deze vragenlijst is om in te vullen nadat je twee dagdelen het spel hebt gespeeld.

Naam:
(die je voor het spel hebt gebruikt)

Geboortedatum:

1. Welke studie doe je/ heb je gedaan?

2. Welk werk doe je/ heb je gedaan?

3. Geef aan hoeveel ervaring je hebt in de volgende activiteiten:

activiteit:	geen	weinig	redelijk veel	veel
werken met een computer				
het spelen van strategische spellen				
het spelen van mastermind				
het spelen van een soortgelijk spel als mastermind? namelijk:				

4. Geef aan hoeveel je weet van:

	vrijwel niks	weinig	redelijk veel	veel
logica				
epistemische logica (kennislogica)				
theory of mind				
truth-conditional semantics				
pragmatiek				

5. Ik zou wel/ niet willen deelnemen aan een vervolgexperiment.

6. Ik wil wel/ niet op de hoogte worden gebracht van de resultaten van dit onderzoek.

Heel erg bedankt voor je deelname aan dit experiment.