UC Merced
Proceedings of the Annual Meeting of the Cognitive Science Society

Title
Predicting Cognitive Difficulty of the Deductive Mastermind Game with Dynamic Epistemic Logic Models

Permalink
https://escholarship.org/uc/item/93k0k49d

Journal
Proceedings of the Annual Meeting of the Cognitive Science Society, 40(0)

Authors
Zhao, Bonan
van de Pol, Iris
Raijmakers, Maartje
et al.

Publication Date
2018

eScholarship.org Powered by the California Digital Library
University of California

Predicting Cognitive Difficulty of the Deductive Mastermind Game with Dynamic Epistemic Logic Models

Bonan Zhao (zbn.dale@gmail.com), Iris van de Pol (i.p.a.vandepol@uva.nl)
Institute for Logic, Language and Computation, University of Amsterdam

Maartje Raijmakers (m.e.j.raijmakers@uva.nl)
Educational Sciences, Free University Amsterdam; Developmental Psychology, University of Amsterdam

Jakub Szymanik (j.k.szymanik@uva.nl)
Institute for Logic, Language and Computation, University of Amsterdam

Abstract

Deductive Mastermind is a deductive reasoning game that is implemented in the online educational game system Math Garden. A good understanding of the difficulty of Deductive Mastermind game instances is essential for optimizing the learning experience of players. The available empirical difficulty ratings, based on speed and accuracy, provide robust estimations but do not explain why certain game instances are easy or hard. In previous work a logic-based model was proposed that successfully predicted these difficulty ratings. We add to this work by providing a model based on a different logical principle—that of eliminating hypotheses (dynamic epistemic logic) instead of reasoning by cases (analytical tableaux system)—that can predict the empirical difficulty ratings equally well. We show that the informational content of the different feedbacks given in game instances is a core predictor for cognitive difficulty ratings and that this is irrespective of the specific logic used to formalize the game.

Keywords: deductive reasoning; mastermind; educational game; cognitive difficulty; logical analysis; computational modeling; dynamic epistemic logic

Introduction

Deductive reasoning is a crucial skill in everyday life as well as in many professions. Children can train this skill by playing educational games like Deductive Mastermind (DMM), in which a secret code needs to be deduced from reasoning about given clues. This game has been implemented in an online educational game system in the Netherlands, Math Garden (Rekentuin),[1] which has resulted in a large and rich collection of user data: Over 200,000 Dutch primary school students have been using this system to practice their mathematical and logical thinking skills (van der Maas & Nyamsuren, 2017). Math Garden records players' speed and accuracy data in solving the game and uses these to compute difficulty ratings (Klinkenberg, Straatemeier, & van der Maas, 2011). These ratings serve as an empirical indicator of the cognitive difficulty of DMM game instances. Such ratings are important for the game experience because for an optimal training-effect it is essential that players are presented with reasoning tasks of the right difficulty level (Ericsson, 2006).

These empirical ratings provide robust estimations of the cognitive difficulty of game instances but do not themselves explain this difficulty. Theoretical complexity measures of game instances can help to better understanding why certain game instances are easy or hard. Such complexity measures are a promising supplement to empirical ratings because they can improve the categorization of the difficuly of game instances. Computational and logical analysis have proven themselves as useful tools to study combinatorial properties of cognitive tasks in order to categorize them into psychologically plausible difficulty classes (for an overview and examples, see, e.g., Isaac, Szymanik, & Verbrugge, 2014; Geurts, 2003; Kemp & Regier, 2012; Feldman, 2000; van Rooij & Wareham, 2008; Verbrugge & Szymanik, 2018). This approach allows us to formalize a cognitive task and extract parameters of the formalization as indicators of the cognitive difficulty of the task.

In this study, we use dynamic epistemic logic (DEL) to analyze the difficulty of the DMM game. We investigate which parts of the logical structure of the deductive reasoning task can predict the cognitive difficulty of DMM game instances. We propose a model of the DMM game based on dynamic epistemic logic, and we derive difficulty measures using formal aspects of this model. On the basis of results from Gierasimczuk, van der Maas, and Raijmakers (2013) we predicted that the different feedback types in the game would be a core predictor for our model, as it was for their analytical tableaux model. In DMM, players are presented with clues that consist of conjectures and corresponding feedbacks. This feedback can be of different types that give different kinds of information, like, "right color but wrong position" or "right color and right position." Our prediction about the importance of the different feedback types was confirmed by our results. The basic features of the DMM game could only explain 27 percent of the variance in difficulty ratings, and adding the DEL measures that did not parameterize over different feedback types only explained up to 43 percent. Including the DEL measures that did parameterize over different feedback types increased the explained variance to 67 percent.

We compare our results with those of Gierasimczuk et al. (2013), who used a model based on a different logical tool: the analytic tableaux system, a proof-theoretic method that uses search trees. Their model is based on the principle of reasoning by cases, and, in addition, it parameterizes over the different feedback types. Similarly to our model, it success-

[1] More information can be found at mathsgarden.com or rekentuin.nl.

fully predicted 66 percent of the variance in the difficulty ratings.[2] The tableaux model builds a search tree to generate all possible cases given the clues and then searches through the tree to find which unique case leads to a solution and which cases are inconsistent. To make predictions about human reasoning, the tableaux model extends the tableaux method with an assumption, based on the properties of different feedback types, about the order of processing the clues in the game. The complexity measures defined on the basis of this model depend on its underlying assumptions about the specific reasoning process of players: the assumption of processing clues one by one and in a specific order, and the assumption that players reason by cases (building up and searching through a tree, also in a specific order), via the tableaux method.

We hypothesize, however, that the predictive power of the tableaux model is independent of these assumptions. We suspect that this model captures something about the underlying structure of the reasoning task that is essential in determining the cognitive difficulty and that the core determiner for this difficulty lies in the different feedback types in the clues in the game. We test this using a model that is based on a different logical system, namely dynamic epistemic logic.

Our DEL model works via the principle of starting from the space of all possible solutions and eliminating answers by updating with the information given by the clues. We present both an order-dependent and an order-independent model that use sequential or simultaneous updates, respectively (see Figures 2 and 3). We pitch our model at Marr's computational level (Marr, 1982), in the sense that it is meant to capture the structure or nature of the reasoning task and makes no commitments about the kind of algorithm or process used to solve it. Since we found that the tableaux and the DEL models have similar predictive power with respect to the cognitive difficulty ratings and moreover we found that their complexity measures are highly correlated, our results imply that although these models use a different formalism, they are tapping into the same underlying structure of the deductive reasoning task.

The Deductive Mastermind Game

Mastermind is played by two players: a code-maker and a code-breaker. The code-maker chooses a sequence of ℓ color pegs (also called pins): the secret code. Each round the code-breaker makes a conjecture about the code by choosing a sequence of ℓ color pegs. The code-maker provides feedback about this conjecture: a black pin for each peg that is of the correct color in the correct position and a white pin for each peg that is of the correct color but in an incorrect position. Based on this feedback the code-breaker places a new conjecture in the next round. Finally, the code-breaker wins the game if she finds the secret code within m rounds.

[2]For the dataset from 2012 that Gierasimczuk et al. (2013) used—containing 100 game instances—the tableaux model predicted up to 75 percent of the variance in difficulty ratings. For the dataset from 2017—containing 355 game instances—it predicted up to 66 percent of the variance.

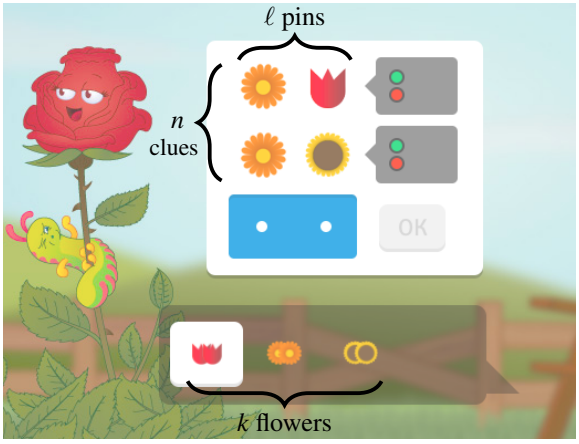

Figure 1: Screen shot of an example DMM game instance

Deductive Mastermind, or Flowercode, as it is called in Math Garden, is a one-player game where, instead of coming up with conjectures, the player is given a sequence of clues. In Math Garden, instead of color pegs, different types of flowers are used to make the game more attractive for children. A game instance consists of k possible flower types and a sequence of n clues, which consist of conjectures, sequences of ℓ flower pins, and corresponding feedbacks, sequences of ℓ feedback pins. Each feedback pin in a feedback corresponds to exactly one of the flower pins in the conjecture. Deducing which feedback pin corresponds to which flower pin is part of the game. The order in which the feedback pins are placed have no meaning. The possible feedback pins that may be used are green (`g`), for a correct flower in the correct position; orange (`o`), for a correct flower in the wrong position; and red (`r`), for flowers that do not occur in the secret code. The game instances are designed in such a way that there exists exactly one answer, one code, that is consistent with the clues. The goal of the player is to deduce this secret code in one go. See Figure 1 for an example of a game instance.

Math Garden offers game instances ranging from 2-pin to 5-pin games. We call a game instance with a secret code of length ℓ an ℓ-pin game. In this paper, we focus on modeling the 2-pin games. The fact that the 2-pin games are the most played instances and that they cover a wide range of difficulty ratings justifies this restriction. The 2-pin games have conjectures and corresponding feedbacks of length 2. We call a sequence of ℓ feedback pins an ℓ-pin feedback. Since the order of the feedback pins have no meaning, there are six distinct 2-pin feedback types: `oo`, `rr`, `gr`, `or`, `gg`, and `go`. Feedback type `gg` is ruled out because it would give away the secret code and feedback type `go` is ruled out because it is inconsistent with a secret code of length 2. Therefore, the allowed feedback types for 2-pin game instances are `oo`, `rr`, `gr`, and `or`.

By means of a computerized adaptive practice system, Math Garden calculates an empirical estimation of how difficult a DMM game instance is to solve, based on players' speed and accuracy data (Klinkenberg et al., 2011; Maris &

van der Maas, 2012). This system uses the following rating principle: The more players that can solve a game instance correctly in a shorter period of time, the easier this game instance is, and *vice versa*. The calculation of these ratings are based on the Elo rating system, which is widely used for calculating capability rankings, such as for chess players (Elo, 1978). Math Garden extends the Elo rating system by, in addition to outcomes, also taking into account reaction time.

Dynamic Epistemic Logic Model

We present a model of the Deductive Mastermind game, using dynamic epistemic logic (DEL). This model is based on the principle of eliminating informational states by updating an initial epistemic model with new information. These informational states are represented by a collection of nodes, called possible worlds. In each possible world certain propositions are set to true or false. The truth values of propositional sentences are evaluated relative to the propositional information that is distributed over the possible worlds.

Dynamic epistemic logic is a particular kind of modal logic (see, e.g., van Ditmarsch, van der Hoek, & Kooi, 2008).[3] Given a set of propositions P, an *epistemic model* $\mathbf{S} = (S, ||\cdot||)$ is a tuple consisting of a set S of possible worlds and a valuation function $||\cdot|| : P \to \mathcal{P}(S)$ that defines the truth values of the propositions in the possible worlds. A change in the information represented by an epistemic model can be represented by an event model. An *event model* $\mathbf{E} = (E, \mathtt{pre})$ is a tuple consisting of a set of events E and a function $\mathtt{pre}$ that assigns a precondition $\mathtt{pre}_i$, some propositional sentence, to each event $e_i \in E$. An epistemic model can be updated by an event model by using the *update operator* $\otimes$, which selects those worlds that satisfy the preconditions of an events (i.e., those worlds that are consistent with the event). The updated model $\mathbf{S} \otimes \mathbf{E} = (S \otimes E, ||\cdot||)$ is an epistemic model with a set of worlds $S \otimes E = \{(s,e) \in S \times E \mid s \models \mathtt{pre}_e\}$ and a valuation function such that $||p||_{\mathbf{S}\otimes\mathbf{E}} := \{(s,e) \in S \otimes E \mid s \in ||p||_{\mathbf{S}}\}$.

Given a game instance with n clues, we model the DMM game as follows. We start from the space of all possible answers, which are all flower sequences of the correct length with flowers of the allowed flower types. This space is determined by the number of available flower types k, and the length of the secret code ℓ (as mentioned earlier, here we model the case that $\ell = 2$). We model this space by an epistemic model $\mathbf{S}_0 = (S, ||\cdot||)$ in which each possible world represents exactly one possible flower sequence and all of the flower sequences are represented by a possible world. See $\mathbf{S}_0$ in Figure 2 for an illustration. We represent the flower types and their position in the flower sequence by means of indexed propositions.[4] We will refer to some flower sequence *a-b* by sentence $a_1 \wedge b_2$ (read: flower a at position 1 and flower b at position 2). Consider the following example with a sunflower and a daisy. Let s stand for sunflower and d for daisy. The flower sequence *sunflower-daisy* is represented in some world w by setting propositions s_1 and d_2 to true in world w and setting all other propositions to false. Then the sentence $s_1 \wedge d_2$ is true in w.

Next, we continue with the clues in the game instance. The feedback given on the flower sequence in a clue limits the number of possible answers; a clue shrinks the space of possible answers to those that are consistent with the clue. The game instances of DMM are designed in such a way that after taking into account all the clues, there is exactly one possible answer left. We translate the informational content (i.e., the eliminative power), of the different feedback types `oo`, `rr`, `gr`, and `or` into preconditions of events in event models. When updating the initial epistemic model with an event model that corresponds to some clue C_i, these preconditions will select only those worlds that represent flower sequences that are consistent with clue C_i.

We represent each clue C_i in the game instance by an event model $\mathbf{E}_i = (E, \mathtt{pre})$. Each event model consists of a single event $e_i \in E$ with a corresponding precondition $\mathtt{pre}_i$. We define the preconditions of these events as follows. Consider a clue C_i consisting of flower sequence $a_1 \wedge b_2$ and feedback type σ. We let

$$
\begin{array}{ll}
\mathtt{pre}_i = b_1 \wedge a_2, & \text{for } \sigma = \mathtt{oo};\\
\mathtt{pre}_i = \neg a_1 \wedge \neg a_2 \wedge \neg b_1 \wedge \neg b_2, & \text{for } \sigma = \mathtt{rr};\\
\mathtt{pre}_i = (a_1 \wedge \neg b_2) \vee (\neg a_1 \wedge b_2), & \text{for } \sigma = \mathtt{gr};\\
\mathtt{pre}_i = (\neg a_1 \wedge \neg b_1 \wedge a_2) \vee (b_1 \wedge \neg a_2 \wedge \neg b_2), & \text{for } \sigma = \mathtt{or}.
\end{array}
$$

The corresponding precondition $\mathtt{pre}_i$ for feedback type `oo` in clue C_i ensures that $\mathtt{pre}_i$ is true in worlds corresponding to flower sequences in which the positions of the two flowers are switched in comparison to the flower sequence in C_i. The precondition $\mathtt{pre}_i$ for `rr` ensures that $\mathtt{pre}_i$ is true in worlds corresponding to flower sequences in which neither of the flower types in C_i occur. The precondition $\mathtt{pre}_i$ for `gr` ensures that $\mathtt{pre}_i$ is true in worlds corresponding to flower sequences in which one of the flowers in C_i is at the right position and the other flower in C_i does not occur. Finally, the precondition $\mathtt{pre}_i$ for `or` ensures that $\mathtt{pre}_i$ is true in worlds corresponding to flower sequences in which one of the flowers in C_i occurs at a different position, and the other flower does not occur.

By updating the initial model $\mathbf{S}_0$ with event model $\mathbf{E}_1$, we get epistemic model $\mathbf{S}_1 = \mathbf{S}_0 \otimes \mathbf{E}_1$. Epistemic model $\mathbf{S}_1$ represents the space of solutions that are consistent with clue C_1. Then in turn, we can update model $\mathbf{S}_1$ with clue C_2 to get epistemic model $\mathbf{S}_2 = \mathbf{S}_0 \otimes \mathbf{E}_1 \otimes \mathbf{E}_2$, which represents the

[3] For readers that are familiar with the details of DEL it suffices to know that we use the basic propositional language, the product update rule, and sphere semantics. This semantics differs from the standard Kripke semantics for epistemic models, by not having a relation over the set of worlds. Furthermore, we use a simplified version of epistemic models and event models, by only using non-pointed models and event models with one event.

[4] Technically, this works as follows. Let $t_1, \ldots, t_k$ be the allowed flower types. Then for each $i \in \{1, \ldots, k\}$ and each $j \in \{1, 2\}$ we define proposition $p_{i,j}$. Proposition $p_{i,j}$ represents that flower t_i is at position j of the flower sequence. The valuation function $||\cdot||$ is defined in such a way that for each flower sequence $(p_{i_1,1}, p_{i_2,2})$, with $i_1, i_2 \in \{1, \ldots, k\}$, there is exactly one world w in the epistemic model such that propositions $p_{i_1,1}$ and $p_{i_2,2}$ are true in this world, and every other proposition is false in this world.

space of solutions that are consistent with clue C_1 and clue C_2. Then $\mathbf{S}_n = \mathbf{S}_0 \otimes \mathbf{E}_1 \otimes \cdots \otimes \mathbf{E}_n$ represents the space of solutions that are consistent with all clues $C_1, \ldots, C_n$. By construction of the game instance, model $\mathbf{S}_n$ consists of exactly one world, which represents the secret flower code. For an illustration, see the example in Figure 2.

We can now represent a game instance with n clues by epistemic model $\mathbf{S}_0$ and event models $\mathbf{E}_1, \ldots, \mathbf{E}_n$. Solving the game then means answering the question of which flower sequence remains in the final updated model $\mathbf{S}_0 \otimes \mathbf{E}_1 \otimes \cdots \otimes \mathbf{E}_n$. This means asking which sentence $\varphi_1 \wedge \cdots \wedge \varphi_l$, representing a possible flower sequence, is true in the final updated model $\mathbf{S}_n$.

Sequential and Parallel update series We use two different update series for the model, namely an order-dependent sequential update and an order-independent parallel update. The sequential update series updates the initial epistemic model with the event models for the clues sequentially, in the top-to-bottom order of the given clues. For $i \in \{1, ..., n\}$ each updated model $\mathbf{S}_i$ is defined as $\mathbf{S}_i = \mathbf{S}_{i-1} \otimes \mathbf{E}_i$. An example is shown in Figure 2. For the parallel update series, each updated model $\mathbf{S}_i$ is defined as $\mathbf{S}_i = \mathbf{S}_0 \otimes \mathbf{E}_i$. This gives a series of updated models $\mathbf{S}_1, ..., \mathbf{S}_n$ of which the intersection is equal to sequential-update model $\mathbf{S}_n$. An example is shown in Figure 3.

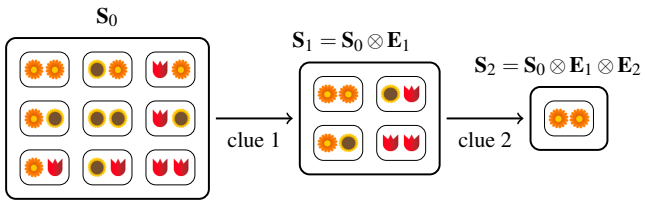

Figure 2: The linear update series for the example in Fig. 1

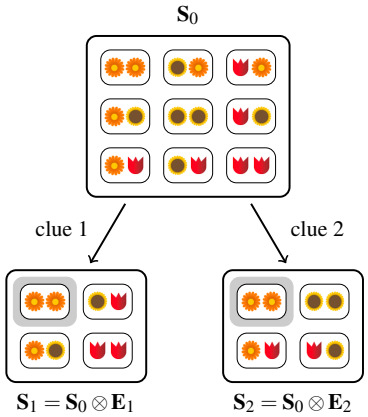

Figure 3: The parallel update series for the example in Fig. 1

Complexity Measures

We now define several complexity measures over the update series generated by the DEL model for DMM game instances.

Size of epistemic models A natural parameter of epistemic models is their size, i.e., the number of worlds in these models. We define the size $|\mathbf{S}|$ of an epistemic model $\mathbf{S}$ as the number of worlds in $\mathbf{S}$, i.e., when $\mathbf{S} = (S, ||\cdot||)$, we have that $|\mathbf{S}| = |S|$. The size of an epistemic model reflects the number of possible answers and therefore the amount of uncertainty that remains.

Average size of epistemic models We define the sum of the epistemic models in a sequential update series $\mathbf{S}_0, \ldots, \mathbf{S}_n$ by $\mathrm{SUM}(\mathbf{S}_0, \ldots, \mathbf{S}_n) := \sum_{i=0}^{n} |S_i|$. Then we define the average size of the epistemic models by $\mathrm{SV}(\mathbf{S}_0, \ldots, \mathbf{S}_n) := \mathrm{SUM}(\mathbf{S}_0, \ldots, \mathbf{S}_n)/n$. The higher the value of this measure, the longer it is the case that many worlds remain in the epistemic model after updating with the clues—the number of clues being equal.

Convergence rate We define the complexity measure CR of a sequential update series $\mathbf{S}_0, \ldots, \mathbf{S}_n$ by the average ratio $|\mathbf{S}_i|/|\mathbf{S}_{i-1}|$ for $i \in \{1, \ldots, n\}$: $\mathrm{CR}(\mathbf{S}_0, \ldots, \mathbf{S}_n) := \sum_{i=1}^{n} (|S_i|/|S_{i-1}|)/n$. The higher the value of this measure, the more difference in informational value between the clues.

Size of epistemic models per feedback type We define the complexity measure *FB-s* of a parallel update series $\mathbf{S}_0, \ldots, \mathbf{S}_n$. This complexity measure is parameterized over the different feedback types and in fact consists of four measures—one for each feedback type $\sigma \in \{\texttt{oo}, \texttt{rr}, \texttt{gr}, \texttt{or}\}$. For each feedback type σ and for each clue C_i that contains σ, we consider the size $|\mathbf{S}_i|$ of the updated model $\mathbf{S}_i = \mathbf{S}_0 \otimes \mathbf{E}_i$. The value of the measure for σ is then defined as the average of $|\mathbf{S}_i|$ for all clues containing σ. If there is no clue containing σ, we give the measure for σ the value 0.

Convergence rate per feedback type Furthermore, we define the complexity measure *FB-r* of a parallel update series $\mathbf{S}_0, \ldots, \mathbf{S}_n$. For each feedback type σ, and for each clue C_i that contains σ, we compute the ratio $|\mathbf{S}_i|/|\mathbf{S}_0|$. The value of the measure for σ is then defined as the average of $|\mathbf{S}_i|/|\mathbf{S}_0|$ for all clues containing σ. If there is no clue containing σ, we give the measure for σ the value 0.

The higher the value of these measures per feedback type, the more worlds remain in the epistemic model after updating with the clues.

Results

For the statistical analysis we used the ratings based on Math Garden user data between November 2010 and April 2017. These data contain 355 DMM game instances with 2 pins. From these 355 instances, 11 instances involved 2 flower types, 82 instances involved 3 flower types, 127 instances involved 4 flower types and 135 instances involved 5 flower types. We tested our complexity measures on this dataset. We computed the value of the complexity measures based on our model for all 355 game instances and used multiple linear regression to see how well our dynamic epistemic logic (DEL) model predicts the variance in the empirical difficulty ratings for these items.

We consider six different regression models. First, Model 0, is a simple model that only includes basic characteristics of the game: the number of flower types, the num-

ber of clues, and whether all flower types are used in the clues. Model 0 explained 27 percent of the variance in difficulty ratings. Model DEL_{SV} extends Model 0 with complexity measure SV, and it slightly improved on Model 0 by explaining 36 percent of the variance. Model DEL_{CR} extends Model 0 with complexity measure CR, and it slightly improved the predictions by explaining 43 percent of the variance. Model $\text{DEL}_{FB\text{-}s}$ extends model 0 with complexity measure *FB-s*, and with 63 percent it explained much more of the variance. Model $\text{DEL}_{FB\text{-}r}$ extends model 0 with complexity measure *FB-r*, and with 67 percent it explained the most variance. So only the measures that parameterize over the different feedback types provided a nice fit. See Table 1 for an overview of the parameter estimates.

Comparison with the tableaux model Furthermore we included Model TABL, which is the regression model used by Gierasimczuk et al. (2013). Model TABL extends Model 0 with complexity measures per feedback type, based on the tableaux model. These measures count the number of nodes in the minimal search tree that is generates from processing the feedbacks in the order `oo,rr,gr,or`. For more details see Gierasimczuk et al. (2013). Run on the data from 2017 their model explained 66 percent of the variance.

Additionally, we ran a regression for the combined model including the measures from both $\text{DEL}_{FB\text{-}r}$ and TABL. With $R^2 = .68$ this combined model did not explain any additional variance (see Table 1). Furthermore, we compared the feedback measures of the Tableaux and the $\text{DEL}_{FB\text{-}s}$ model and we found high correlations (see Table 2).

Discussion

We investigated the difficulty of Deductive Mastermind (DMM) game instances with tools from dynamic epistemic logic (DEL). We proposed a formalization of DMM, in which we used epistemic models to represent possible answers and event models to encode the information in the clues. Based on parameters of this model we formulated several complexity measures to capture the difficulty of game instances. Our model was able to successfully predict 67 percent of the variance in the empirical difficulty ratings. Including our complexity measures in the regression model greatly increased the fit in comparison to the simple model that uses only basic characteristics. These findings show that the dynamic epistemic logic modeling method has merit.

When comparing the different complexity measures that we used it is noteworthy that only the complexity measures that parameterize over the four different feedback types gave a good fit. The complexity measures that did not parameterize over feedback types were not even able to explain half of the variance, while the complexity measures that did parameterize over feedback types explained two third of the variance. This confirms our prediction, based on the results by Gierasimczuk et al. (2013), that the informational content of the different feedback types is a core determiner of the cognitive difficulty of solving a DMM game instance.

We compared our DEL model with the tableaux model by Gierasimczuk et al. (2013), which also parameterizes over the different feedback types. The DEL model and the tableaux model measure different aspects of the DMM game. The tableaux model builds a search tree, generating all possible cases, and searches through this tree to find the unique case that leads to a consistent answer. It measures the length of the search path in the minimal search tree, per applied feedback type. The DEL model, on the other hand, focuses on the space of possible solutions and it measures how the size of this space shrinks by the information in the clues.

Despite the differences in the construction of the two models, their results were very similar. Combining the tableaux model with the DEL model did not explain any additional variance, and we found high correlations between their complexity measures. These results imply that the tableaux and the DEL models capture an essential part of the structure of the DMM reasoning task and that their predictive power is independent of the specific formalization, i.e., the specific type of logic, that is used. These results also show that the predictive power of the tableaux model is not dependent on its assumptions about processing the game in terms of reasoning by cases, or the fact that the model is order dependent—since our DEL model is order independent, using a parallel update, and does not use reasoning by cases.

An aspect that both the DEL and tableaux model can improve on is their restriction to 2-pin games. Future research may include extending these models to games with codes of lengths 3, 4, and 5, to predict the variance in difficulty ratings for all instances of the DMM game in Math Garden. To gain further insight in the kind of reasoning used in DMM, it is interesting to look at error patterns in responses. Therefore, future research may also include investigating and explaining these patterns in terms of formal aspects of the logical structure of game instances. In addition to this, future research could look at the learning patterns of successful answers. For certain game instances (like game instances with only `gr` feedbacks, such as shown in the example in Figure 1) players seem to be learning shortcuts. The DEL model could be used to investigate whether logical shortcuts based on cross-clue reasoning can explain such learning patterns.

In this paper, we showed that logic-based modeling methods can be used successfully to predict the cognitive difficulty of deductive reasoning tasks. We believe that similar techniques to the one developed in this paper can be used to better understand factors contributing to the cognitive difficulty of a variety of other cognitive tasks. With this study we hope to contribute to a growing body of work that shows that computational models based on logical principles can be of psychological relevance for investigating human reasoning, such as applied in deductive reasoning games.

Acknowledgments

We thank Oefenweb and Han van der Maas for sharing data from Math Garden and Nina Gierasimczuk for sharing the

Table 1: Parameter estimates of the DEL and tableaux regression models

	Model 0	DEL_{SV}	DEL_{CR}	$DEL_{FB\text{-}s}$	$DEL_{FB\text{-}r}$	TABL	$DEL_{FB\text{-}r}$+TABL
(Intercept)	−17.8894***	−27.7091***	−42.2720***	−10.28256***	−26.8712***	−14.432994***	−10.1257***
#flower types	−0.3354	20.7143***	−5.7883***	2.47430***	34.6568***	4.562682***	−2.3511***
#clues	4.5300***	−4.5951***	58.0237***	−2.16959***	−7.6840***	−1.016774*	1.7511*
allflowersinitem	−8.8332***	−6.3370***	−6.4818***	−5.02605***	−6.9753***	−6.089445***	−5.4334***
SV		−3.3568***					
CV			−75.8919***				
ooD				−8.98402***	−55.209***		−8.3863**
rrD				−0.04604	−46.610***		0.2147*
grD				0.70094***	−41.0523***		0.8798***
orD				1.83475***	−21.3221***		0.7604**
ooT						−11.455507***	0.5752
rrT						−2.983262***	−0.9595
grT						0.003135	0.4168*
orT						2.518258***	2.4814**
R^2	0.2679	0.3581	0.425	0.6322	0.672	0.6614	0.6847
Num. obs.	355	355	355	355	355	355	355

The measures for ooD, rrD, grD, and orD are defined by *FB-s* and *FB-s*, for $DEL_{FB\text{-}s}$ and $DEL_{FB\text{-}r}$, respectively; the measures for ooT, rrT, grT, and orT are defined by the tableaux model (corresponding to Model 1 in Gierasimczuk et al., 2013).

*** $p < 0.001$, ** $p < 0.01$, * $p < 0.05$

Table 2: Correlations between the feedback measures of the Tableau Model (T) and the DEL model (D)

	ooD	rrD	*grD*	*orD*
ooT	0.9642	−0.0763	−0.2312	−0.2843
rrT	−0.0720	0.7475	−0.0012	−0.2177
grT	−0.3165	−0.0580	0.6448	−0.1270
orT	−0.2674	−0.1293	−0.2096	0.7632

code of the tableaux model presented in Gierasimczuk et al. (2013). We thank the members of the Computational Cognitive Science group at the Radboud University for helpful discussion and Ronald de Haan for graphical and technical support. We thank four anonymous reviewers for their helpful feedback. This paper is based on the MSc thesis research of BZ (Zhao, 2017). IvdP was supported by Gravitation Grant 024.001.006 of the Language in Interaction Consortium from the Netherlands Organization for Scientific Research (NWO). JS was supported by the ERC under the European Union's Seventh Framework Programme (FP/2007–2013)/ERC Grant Agreement n. STG 716230 CoSaQ.

References

Elo, A. (1978). *The rating of chessplayers, past and present*. Arco Pub.

Ericsson, K. A. (2006). The influence of experience and deliberate practice on the development of superior expert performance. *The Cambridge handbook of expertise and expert performance*, *38*, 685–705.

Feldman, J. (2000). Minimization of boolean complexity in human concept learning. *Nature*, *407*(6804), 630–633.

Geurts, B. (2003). Reasoning with quantifiers. *Cognition*, *86*(3), 223–251.

Gierasimczuk, N., van der Maas, H. L. J., & Raijmakers, M. (2013). An analytic tableaux model for deductive mastermind empirically tested with a massively used online learning system. *Journal of Logic, Language and Information*, *22*(3), 297–314.

Isaac, A., Szymanik, J., & Verbrugge, R. (2014). Logic and complexity in cognitive science. In *Johan van Benthem on logic and information dynamics* (pp. 787–824). Springer.

Kemp, C., & Regier, T. (2012). Kinship categories across languages reflect general communicative principles. *Science*, *336*(6084), 1049–1054.

Klinkenberg, S., Straatemeier, M., & van der Maas, H. L. (2011). Computer adaptive practice of maths ability using a new item response model for on the fly ability and difficulty estimation. *Computers & Education*, *57*(2), 1813–1824.

Maris, G., & van der Maas, H. L. (2012). Speed-accuracy response models: Scoring rules based on response time and accuracy. *Psychometrika*, *77*(4), 615–633.

Marr, D. (1982). Vision: A computational investigation into the human representation and processing of visual information.

van der Maas, H. L., & Nyamsuren, E. (2017). Cognitive analysis of educational games: The number game. *Topics in cognitive science*, *9*(2), 395–412.

van Ditmarsch, H., van der Hoek, W., & Kooi, B. (2008). *Dynamic epistemic logic* (Vol. 337). Springer.

van Rooij, I., & Wareham, T. (2008). Parameterized complexity in cognitive modeling: Foundations, applications and opportunities. *The Computer Journal*, *51*(3), 385–404.

Verbrugge, R., & Szymanik, J. (2018). Tractability and the computational mind. In *Handbook of the computational mind.* Routledge. (forthcoming)

Zhao, B. (2017). *Dynamic Epistemic Logic Models for Predicting the Cognitive Difficulty of the Deductive Mastermind Game*. Unpublished master's thesis, University of Amsterdam.