

CHALMERS



Anthropomorphic Proof System for First-Order Logic

Master of Science Thesis in Intelligent Systems Design

ABDUL RAHIM NIZAMANI

Department of Applied Information Technology
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden, 2010
Report No. 2010:128
ISSN: 1651-4769

Anthropomorphic Proof System for First-Order Logic

Copyright © Abdul Rahim Nizamani, 2010

Report No. 2010:128

ISSN 1651-4769

Examiner: Claes Strannegård

Supervisor: Claes Strannegård

Department of Applied Information Technology

Chalmers University of Technology

Göteborg, Sweden

Email: abdulra@chalmers.se

Göteborg, Sweden 2010

Abstract

This thesis presents a computer-based experiment conducted to study the human reasoning in first-order logic. Fifty questions of FOL with accompanying graph models were presented to participants, who were asked to determine which questions were true and vice versa. The questions appeared randomly and with a fixed time limit, and the response times were recorded for all the right answers. A suggested proof system for FOL with bounded cognitive resources is analyzed using the results from this experiment.

Results show that the proposed proof system models the human reasoning much better for some problems but not for all. With some shortcomings and space for improvement, the system gives a working concept for better proof formalisms with limited cognitive resources, which can generate proofs that are easier to comprehend than the purely mathematical proof strategies used today.

Acknowledgements

First of all, I thank my supervisor Claes Strannegård who trusted me for this research work and entrusted to me the work in his project of anthropomorphic artificial intelligence at the IT University of Göteborg. This project is in continuation to his previous works in the same area, and indeed the work presented herein is based on his proposed topic.

Next, I thank the participants in this research work, including Fredrik Engström and Simon Ulfsbäcker.

I am grateful to Lance Rips, professor of cognitive psychology, NorthWestern University USA, who helped us in the design of the experiment.

I am also grateful to all the participants in the experiments with whose volunteer help we have been able to produce the results. Without them, this project could never be realised.

Contents

Abstract	ii
Acknowledgements	ii
1 Introduction	1
1.1 Anthropomorphic Artificial Intelligence	1
1.2 Thesis Outline	2
2 Background	3
2.1 Cognitive Modeling	3
2.2 Logical Reasoning	4
2.3 Previous Work	4
3 Proof Formalism	5
3.1 Formula	5
3.2 Proof Systems	6
3.3 Proof Rules	6
3.4 Complexity Measures	6
3.4.1 Formula Measures	6
3.4.2 Model Measures	7
3.4.3 Proof Measures	8
3.5 Proof Finder	8
4 Experiments	10
4.1 Preliminary Experiments	10
4.2 Controlled Experiments	10
4.2.1 Participants	10
4.2.2 Questions	11
4.3 User Interface	12
4.3.1 Procedure	13

4.4	Measures	13
5	Results	16
5.1	Results of the experiment	16
5.1.1	Accuracy	17
5.1.2	Latency	17
5.2	Proofs	17
5.2.1	Usage of Rules	17
5.3	Correlation	17
6	Discussion	21
6.1	Interviews with Participants	21
6.2	Formula and Model Complexities	21
6.3	Multiple Correlation	22
6.3.1	Model Measures	22
6.3.2	Formula Measures	23
6.4	Selective Regression	23
6.4.1	Selecting by Higher accuracy	23
6.4.2	Selecting by Formula Complexity	24
7	Conclusion	26
7.1	Summary and Conclusion	26
7.2	Future Work	26
	References	27
A	List of Test questions	29
B	Models for the Test Questions	31
C	Test Results	37
D	Axioms	41
D.1	Tautologies	41
D.2	Contradictions	44
D.3	Non-contradictions	45

Chapter 1

Introduction

1.1 Anthropomorphic Artificial Intelligence

Since the advent of computers, artificial intelligence has remained a major area of research in computer science. Thousands of scientists have devoted their precious time to realise some of the important fantasies of human mind by the help of computers. The charm of artificial intelligence is not limited to scientists only, as much of the science fiction fantasy circulates around advanced topics in A.I.

Most of the A.I. research is dedicated to the computational modeling of human reasoning. Computers can process complex problems and data by using mathematical models of human learning and reasoning processes. But as a matter of fact, the prevalent A.I. research only considers the strengths of human mind and memory, and builds upon that to design A.I. applications that can beat humans in complex problems such as chess playing. Output of such systems is too complex to be understood or consumed by humans. There are certain classes of problems where the computer-generated output is usually meant to be consumed by humans. Thus such an A.I. application will have to consider not only the strengths of human reasoning but also the weaknesses and limitations of the brain. This class of A.I. programs is termed as *Anthropomorphic Artificial Intelligence*.^[1]

Indeed, this approach to A.I. that considers limitations of human mind and memory, could also be helpful in solving very complex problems at which humans are still better than computers. The Chinese game of Go is a typical example in which today's supercomputers are still not able to beat the best human players. The game has a vast search space of possible moves, too huge to be explored by computers. Humans tend to overcome this problem, possibly by ignoring many possibilities that the computer will usually explore.

This thesis presents the analysis of a proof system for first order logic that works with bounded cognitive resources. This proof formalism is developed by Claes Strannegård and his research fellows including myself, at the Department of Applied I.T., Göteborg University, Sweden. The complete system is yet to be published. This thesis only concerns the details of the experiment to analyze this new model and presents its results. An analysis of this proof formalism is carried out using a psychological experiment carried out in the same

department, in order to assess its performance and validate its results. The work builds on earlier works by Claes et.al. in [1, 4].

This report only presents my part of the work except where stated otherwise.

1.2 Thesis Outline

Chapter 2 presents the background for this thesis including key concepts in cognitive modeling and logical reasoning. Related work is discussed in the last section.

Chapter 3 discusses the computational model that is analyzed in this thesis. This model is developed by Claes Strannegård, myself and other members of our team.

Chapter 4 presents the design and conduct of the experiment to study the human reasoning in first-order logic. The results of this experiment, presented in chapter 5, are used to analyze the proposed computational model for proving the truth of FOL models.

Chapter 6 discusses the results gathered from the experiment, examines the proposed proof formalism and presents some important findings. The thesis is concluded in chapter 7 with suggested future work.

Chapter 2

Background

2.1 Cognitive Modeling

In cognitive science, the human mind is often seen as an information processor, with its various memory systems to store and process information. Many computational models used in cognitive psychology are presented in [10].

Atkinson and Shiffrin proposed a basic memory model in 1968 with three parts of human memory. They divided memory in three stages: Sensory Memory (SM), Short-term memory (STM) and Long-term memory (LTM) [11]. This three-stage model was further refined by Park and Gutchess, who divided the LTM into Declarative Memory (DM) and Procedural Memory (PM) [10]. The declarative memory stores semantic information, factual information as well as some personal experiences. The procedural memory stores methods or processes, such as basic arithmetic operations or rules of logic.

Sensory memory stores information from the senses for very short time spans, usually under one second or no more than two. However, information from this memory can be transferred to other memory systems for longer conservation. There are various components of sensory memory, but for the purpose of this thesis, we are more interested in visual memory (VM) that stores incoming visual information. This can store scenes, text, or logic formulas in our case.

Baddeley and Hitch suggested working memory model in 1974, to replace the simplistic notion of short-term memory. Later, it was further refined by Baddeley and others [12]. Working memory (WM) actively holds information, usually for short durations, and supports cognitive functions such as reasoning, comprehension and learning. It is recognized to have four components, the central executive, the phonological loop, the visuospatial sketchpad and the episodic buffer [10].

The central executive acts as a controller for the other components. The phonological loop stores speech and sounds in a subvocal form, and is able to convert the visually presented speech information such as words into its phonological code. The visuospatial sketchpad is a temporary storage for visual and spatial information. The episodic buffer acts as a link between these systems and the LTM.

Working Memory is severely limited and holds small chunks of information at once. Miller suggests that it can store up to seven items of some kind, such as digits or symbols [2]. Later studies show that it is even less than that in general, whereas for some individuals it can even be larger.

2.2 Logical Reasoning

Logical reasoning has been studied in many scientific research areas, and goes back to the times of Aristotle. Everyday logical reasoning is usually quite different from reasoning in classical logic. Most of the experiments conducted in logical reasoning formulate trials in natural language rather than as logic formulas [13]. This can be natural for studies not involving classical logic, but it can give rise to ambiguities when used within the classical logic. Connectives in mathematical logic usually do not exactly correspond to the structures in human language. For example, the "or" in English does not always correspond to the logical connective \vee , as "or" can be exclusive or inclusive but \vee is only inclusive. Similarly the quantifiers in predicate logic do not correspond to their natural language counterparts. Newstead's investigations into syllogisms [14, 15] show that it is quite common in everyday reasoning to jump from the assumption "Some A are B" to the conclusion "Not all A are B". Whereas the quantifier \exists does not lead to $\neg\forall$ in logic.

2.3 Previous Work

Natural deduction systems were developed by Jaskowski [8] and Gentzen [7] in the 1930s. Gentzen's goal was to define a formal proof system for logic that was psychologically realistic and close to the actual human reasoning (page 74 in [7]). Several other formalisms were later developed, many of them derivatives of natural deduction. But none of them follows the memory models developed by Atkinson and Shiffrin and later refined by others, which is necessary to make them close to the actual human reasoning.

This thesis builds on the previous work of Claes, et. al. presented in [5]. They propose a model of human reasoning in propositional logic and analyze it with the help of a psychological experiment. They state that their suggested proof formalism is local as all the successor states depend only on their immediate predecessor state, and linear as the states appear in linear order.

The proof formalism models rules to prove a propositional statement as a tautology or non-tautology, providing two proof systems respectively. An experiment was conducted with student volunteers to study the human reasoning in propositional logic and to validate the suggested proof model. They conclude that their work shows that it is possible to define proof systems that can go beyond natural deduction and incorporate concepts from cognitive psychology, in order for them to be close to human reasoning.

Chapter 3

Proof Formalism

This chapter presents the computational model of the proof formalism which is examined in this thesis. This model was developed by our team headed by Claes Strannegård and is not solely my work. It is based on previous work by Claes et.al. in [1, 4, 5, 9], and is to be published in a later publication with complete details. Here I summarize it for the readers of this thesis.

3.1 Formula

Definition 1. A *term* is any variable or constant. The variables used in the model are x , y and z . The constants are natural numbers, $1..n$, representing the node number in the respective model.

Definition 2. The *formulas* are defined as follows.

- The propositional constants \perp and \top are formulas.
- Propositional variables and abstraction variables are formulas.
- If P is a predicate symbol of arity $n \geq 1$, and if t_1, t_2, \dots, t_n are terms, then $P(t_1, t_2, \dots, t_n)$ is a formula.
- If A is a formula, then so is $\neg A$.
- If A and B are formulas, then so are $(A \vee B)$, $(A \wedge B)$, $(A \rightarrow B)$ and $(A \leftrightarrow B)$.
- If A is a formula and x is a variable, then $(\forall xA)$ and $(\exists xA)$ are also formulas.

Definition 3. The *length* of a formula A denoted by $|A|$ is defined inductively as follows.

- $|\perp| = |\top| = 1$
- For any variable x or a constant c , $|x| = |c| = 1$
- Precates: $|P(t_1, t_2, \dots, t_n)| = n + 1$

- $|\neg A| = |A| + 1$
- Binary Connectives: $|B \diamond C| = |B| + |C| + 1$
- Quantifiers: $|\forall x A| = |\exists x A| = |A| + 2$

3.2 Proof Systems

The proof system consists of two sub systems, one for proving truth and one for falsity. A proof in our system is a trace of the use of four memory types, i.e., declarative memory (DM), visual memory (VM), working memory (WM) and procedural memory (PM). The FOL sentence lies in the working memory where it is reduced by the rules of the proofs which are stated in the PM. Each step is an application of a proof rule to the previous contents of the WM.

Some rules use the declarative memory contents (stored in a file for the proof finder, given in Appendix D). The application of such rules retrieves related contents from the file which are shown in the DM field in the proof.

Many of the fol sentences are longer than the visual memory (set to 6 in our experiments). To overcome the limit, abstraction variables are used to refer to the whole or parts of the formula. Later, the abstraction variables are expanded by looking up the specific part of the formula, hence keeping the VM field within the bound. Any rule requiring the lookup of a part of the formula stores that part into the VM.

Thus, a proof is an application of rules that reduce the original fol sentence to either T or F, depending upon whether the question was True or False. WM depicts the reduction of the sentence with rule application.

3.3 Proof Rules

Table 3.3 lists the basic rules of the proof system. Further rules for the truth system are listed in table 3.3 and for falsity system in table 3.3.

An example of a proof (of problem 4) is presented in table 3.3, generated with proof size = 17 and proof length = 6 steps. It was generated using visual memory size 6 and working memory size 8, while minimizing the proof size. See Appendix B for the model for this problem.

3.4 Complexity Measures

3.4.1 Formula Measures

Formula length is a simple complexity measure defined as the *number of symbols* in the formula excluding the parentheses. The difficulty of the problem is somewhat related to this measure, as the larger formulas are usually harder to solve. Other measures can also be used for the length, such as the number of *binary connectives* or the *number of predicates* in the formula. *Number of quantifiers* also adds to the complexity of a formula, and is a good

Name	Rule	Conditions
Formula Inspect:	$\frac{B(a_i)}{B(C)}$	if $a_i \leftrightarrow C \in VM$ and $f(a_i) = C$, where f is the standard abstraction of A .
Rewrite:	$\frac{B(C)}{B(C')}$	if $C \leftrightarrow C' \in D$ is a tautology, $ C' < C $ and $B(C')$ is obtained by replacing exactly one occurrence of C in B by C' .
\forall Reduce:	$\frac{B(\forall x \in DC(x))}{B(\forall x \in D - iC(x) \wedge C(i))}$	if i is the smallest constant in D .
\exists Reduce:	$\frac{B(\exists x \in DC(x))}{B(\exists x \in D - iC(x) \vee C(i))}$	if $i \in D$.
Delta Inspect:	$\frac{B(C)}{B(v)}$	if $C \in VM$, $C \in I$ and v is the truth value of C in M .
Pi Inspect:	$\frac{B(\forall x \in DC(x))}{B(\forall x \in D - iC(x))}$	if $C(i)$ is quantifier free and true in M .
Sigma Inspect:	$\frac{B(\exists x \in DC(x))}{B(\exists x \in D - iC(x))}$	if $C(i)$ is quantifier free and false in M .
Pi \rightarrow Comprehension:	$\frac{B(\forall x \in D(col(x) \rightarrow C(x)))}{B(\forall x \in D'C(x))}$	where $D' = \{d \in D : col(d) \text{ is true in } M\}$.
Sigma \wedge Comprehension:	$\frac{B(\exists x \in D(col(x) \wedge C(x)))}{B(\exists x \in D'C(x))}$	where $D' = \{d \in D : col(d) \text{ is true in } M\}$.

Table 3.1: Basic System Rules

measure as shown in the results.

One of the complex measures to weigh the formula is the depth of its parse tree. This can be a good measure, as one may argue that a user will have to parse the formula while trying to solve it.

3.4.2 Model Measures

The complexity of the models accompanying the formulas can be measured in simple values such as the *number of nodes* and the *number of edges*. Since the models are directed graphs, another measure can be the *number of connections*, i.e. the number of directed edges. *Number of colors* is also used in the analysis which is the number of unique colors used to color the graph nodes.

Name	Rule	Conditions
Strengthening:	$\frac{A}{A'}$	if $A' \rightarrow A \in DM$ is a tautology.
Tautology Recall:	$\frac{A}{\phi}$	if $A \in DM$ is a tautology.

Table 3.2: Truth System Rules

Name	Rule	Conditions
Weakening:	$\frac{A}{A'}$	if $A \rightarrow A' \in DM$ is a tautology.
Contradiction Recall:	$\frac{A}{\phi}$	if $A \in DM$ is a contradiction.

Table 3.3: Falsity System Rules

3.4.3 Proof Measures

Proof Length

It is defined as the length of a proof using the suggested rules, and is equal to the number of steps to solve a problem.

Proof Size

It is defined as the sum of the lengths of formulas appearing in the proof. Minimum proof size is the smallest proof size of a proof for a given problem and is used as the main complexity measure in this work.

This is mainly because it is a simple measure, but also captures more details of a proof than the proof length. Although like other mathematical complexity measures, it is prone to heavy criticism. But note that it measures the usage of the working memory, making it closer to the psychological complexity of the proof.

Proof Weight

Proof weight is a variation of proof length in which different rules are assigned different weights and the sum of weighted rules is considered as proof weight.

Proof Volume

This is a combination of proof weight and proof size, where the length of each formula in the proof is applied a weight according to the rule used to generate that formula. The sum of the weighted lengths of formulas is defined as the volume of a proof.

3.5 Proof Finder

A program was implemented in Haskell by one of the team members to find the proofs in the proposed proof system. It was continuously updated during the work to implement the latest changes during the development of the system.

DM	VM	WM	PM
	$a0 \Rightarrow x\{1, 2, 3\}a1$	$a0$	Formula Inspect
$a1[x = 3] \rightarrow (\forall x\{1, 2, 3\}a1)$		$\forall x\{1, 2, 3\}a1$	Weakening
	$a1 \Rightarrow y\{1, 2, 3\}a2$	$a1[x = 3]$	Formula Inspect
$a3 \leftrightarrow (\forall y\{1, 2\}a2[x = 3])$		$\forall y\{1, 2, 3\}a2[x = 3]$	For-all Reduce+
		$a3 \wedge a2[x = 3, y = 3]$	Delta Inspect
$a3 \wedge F$		$a3 \wedge F$	Contradictin Recall
		F	

Table 3.4: A proof of problem 4: $\forall x(\forall y(\neg E(x, y) \rightarrow \text{Yellow}(y)))$

I will not describe it as it is not my work, but I used it to generate proofs with varying parameters. It accepted following parameters as command-line options:

```
folp "formula" ltm-file pm-file model-file model (T|NT) SM-size WM-size
(--length | --size | --weight | --volume)
```

where,

- `ltm-file` is the file containing formulas appearing in Appendix D
- `pm-file` is a text file containing names of rules to be used, optionally with a weight to be applied (for proof weight and volume measures)
- `model-file` is a text file containing model structures. It was encoded to encode graph models in text format, in which each line encodes a single node of any graph in the format: `1,1,red,"2,3"`, containing node number, model number, node color and the list of nodes to which it connects, respectively.
- `model`: model number (same as problem number)
- `(T|NT)` for true and non-true formulas
- `SM-size`: the size of visual memory, set to 6 for all problems
- `WM-size`: the size of working memory, set to 8 for all problems
- `(--length | --size | --weight | --volume)`: the proof measure to be used to generate proofs. It generates the proof with minimum value of this measure.

A sample run of the program for problem 4 is as follows:

```
folp ">x>y(~E(x,y)->Yellow(y))" ltm.txt pm.txt models.txt 4 NT 6 8 --size
```

Chapter 4

Experiments

This chapter focuses on the design of the experiment conducted to examine the proposed proof formalism presented in chapter 3. A description of the design of test questions consisting of FOL formulas and graph models is presented, along with the design of web-based user interface.

The proposed proof system is designed according to the memory model of humans, following the human reasoning with limited cognitive resources. This requires cognitive validation of the model, which can be accomplished using a trial of human participants. A high correlation of the test measures with the suggested complexity measures will be indicative of the system's relationship with the human reasoning.

Similar experiments have been conducted previously. The one presented here builds on an earlier very similar test for propositional logic [5]. Other experiments have been conducted to study the use of arithmetic by children [9].

4.1 Preliminary Experiments

A preliminary test was conducted with two student participants. The test consisted of eight unique first order sentences and eight different graph models. Each FOL sentence was tested on each graph, thus making 64 different problems to be solved. The models were hand-crafted simple graphs with colored nodes and undirected edges.

The results showed that the problems were rather too simple, and the fol sentences were being repeated. The repetition allowed the participants to memorize the problems and solve them better in later questions. The final experiment was designed to include unique problems for every question.

4.2 Controlled Experiments

4.2.1 Participants

Ten computer science students were selected from Gothenburg, Sweden, using email invitation. They were from mixed nationalities and in the age span of 20-30 years, with nine of them being men and one woman. All of them had studied first-order logic in their university

studies. Snacks were also offered to them for participating in the experiment.

4.2.2 Questions

Fifty questions were prepared for the final controlled experiment. Each question consisted of a first-order formula accompanied with a model for that formula. The objective was to decide whether the formula is true or not for the given model, i.e. to give the truth of the problem (see example in Figure 4.3 on page 15 for the presentation of a problem). The questions included 24 true formulas and 26 false formulas.

Formulas

The FOL sentences used in the test were manually constructed and were continuously modified to fit the experiment. Following points were considered to build the final set of formulas:

- The final 50 sentences were of varying length, with their formula length from 5 to 19.
- Number of predicates used in each formula was from 1 to 3.
- Number of quantifiers in each sentence was 1 to 3, however only two formulas had three quantifiers. It was observed that an extra quantifier adds considerable complexity to the question even with limited size of the models.
- Same variables were used in all the formulas, x , y and z , and in this specific sequence. Reusing a variable in the same formula was avoided to prevent confusion. For instance, a sentence such as $\forall x \text{Red}(x) \wedge \exists x \text{Blue}(x)$ was rewritten as $\forall x \text{Red}(x) \wedge \exists y \text{Blue}(y)$.
- A negation (\neg) at the start of a formula was avoided. Although it does not change the mathematical complexity (except an extra step of reversing the truth of original formula), it significantly adds to the psychological level of difficulty of the problem.
- Multiple parentheses are hard to parse for humans. In FOL, since parentheses are used for predicates to contain their arguments, groups inside the formulas were bracketed using square brackets or large-sized parentheses to simplify the human reading of the formulas. The after-interviews from the participants also showed that reading and parsing the formula takes much longer than solving it.
- Four predicates were used: $\text{Red}(x)$, $\text{Blue}(x)$, $\text{Yellow}(x)$, all representing the color of node x , and $\text{E}(x, y)$ representing an edge between nodes x and y .

The full list of sentences used in the experiment is provided in Appendix A along with their truth value for the models given in Appendix B.

Models for Formulas Initially, simple undirected graphs with colored nodes were suggested to act as the models for FOL formulas used in the experiments. Graphs are pretty simple models for FOL formulas and an undergraduate computer science student is well

aware of the basic graph theory. To keep the complexity under control, number of nodes and edges was limited.

During the preliminary experiments, the undirected graphs proved to be too simple. To raise the complexity of the graphs, the directed graphs were chosen for the final experiments. To generate 50 graphs with as much randomness and dissimilarity as possible, and having few choices in terms of number of nodes, a simple procedure was written in PHP to generate random graphs. This routine generated Latex code for the graphs. Finally, 50 graphs were hand-picked from the set of generated graphs, mainly due to the fact that not all random graphs were visually clear. Intercrossing edges for example were avoided.

To put some limits, the nodes were limited to 3-4, and the number of edges was constrained in 3-6 including loops. The nodes were colored randomly with red, blue and yellow. The comments by psychology professor Lance Rips showed that about 10% males are color blind for red/green. So green was avoided as red had been used.

The full list of graphs is provided in Appendix B.

4.3 User Interface

The experiment was conducted on the web. The user interface for the test is based on the previous interface initially developed by Jorge Garcia and later modified by Claes et.al. for use in the similar experiment for propositional logic [5]. This open-source design was further modified to accommodate for the first-order logic, specially the presentation of graph models alongwith the logic sentences.

Important design elements including the modifications are as under:

- The web URL lands on a html page containing instructions about the test. A link at the bottom of the page directs the user to the actual test interface.
- The first page of the test interface lists the truth table for Logic operators and a list of the Predicates used in the test (Figure 4.1).
- The second screen requests user particulars including a nickname in order to make the results anonymous and build the user confidence. Age and sex are also requested on this page.
- A question is presented as a model with a logic sentence stating some statement about the model. The user is asked about the truth of the problem, and she answers by evaluating that statement in the accompanied graph. (See figure 4.3).
- Every two consecutive questions are separated with an intermediate screen that allows the user to get ready for the next question by taking a little break.
- A time-out of 90 seconds was implemented. If a user takes longer than that, the current problem is aborted and the user is directed to the intermediate page to start the next question.

- The user is able to skip a problem by clicking on the "I don't know" option. This also helps to avoid guesses by the users, as they are encouraged to skip a question rather than to guess it, in case they are not sure about the answer.

Sample screen shots of the test interface are provided in figures 4.1 (p.13), 4.2 (p.14), 4.3 (p.15) and 4.4 (p.15).

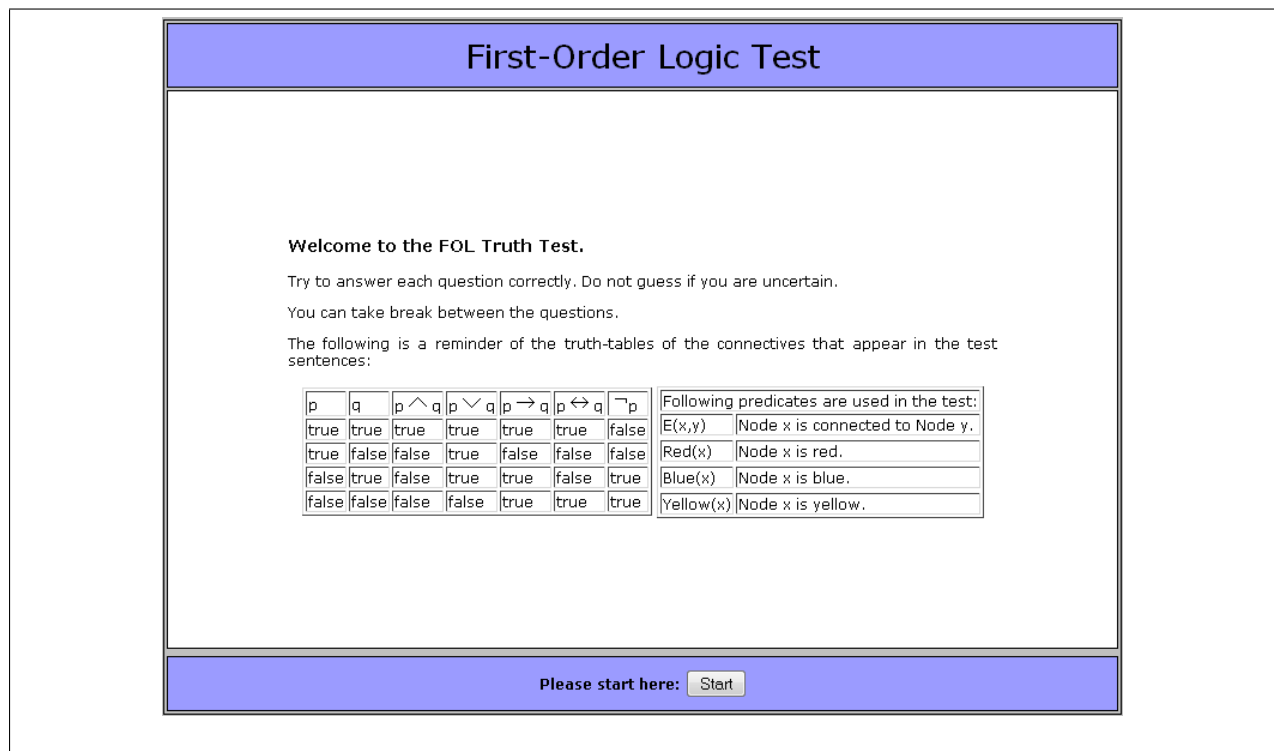


Figure 4.1: Web Interface: Start Screen

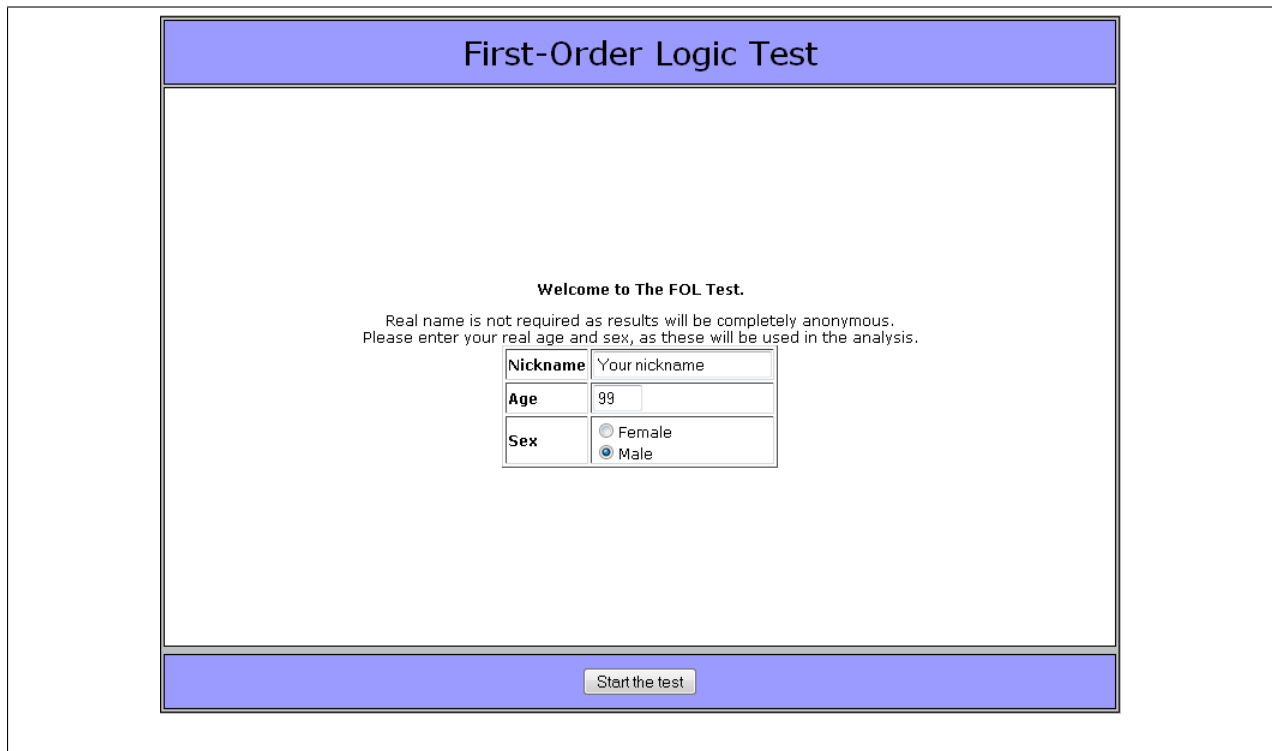
4.3.1 Procedure

The experiment was conducted in a computer laboratory in the I.T. University of Gothenburg, Sweden. The participants were each assigned computer terminals individually. The duration of the experiment was one hour, with a short break after 25 minutes.

After being seated, the participants were given instructions printed on a page which were also read aloud to them. They were first introduced to the sample test in order to familiarize them to the test setup and sample questions. After they were confident to proceed to the actual test, they were guided to the new URL of the real test. After 25 minutes, they were requested to take a break of short time after finishing the question at hand.

4.4 Measures

The test recorded the answers of the participants along with their response times for each answer. The response times were recorded on the client computer and then sent to the server in order to avoid the network delays.



The screenshot shows a web interface titled "First-Order Logic Test". The interface is enclosed in a blue border. At the top, there is a blue header bar with the text "First-Order Logic Test". Below the header, the main content area is white and contains the following text:

Welcome to The FOL Test.
Real name is not required as results will be completely anonymous.
Please enter your real age and sex, as these will be used in the analysis.

Below the text is a registration form with three rows:

Nickname	Your nickname
Age	99
Sex	<input type="radio"/> Female <input checked="" type="radio"/> Male

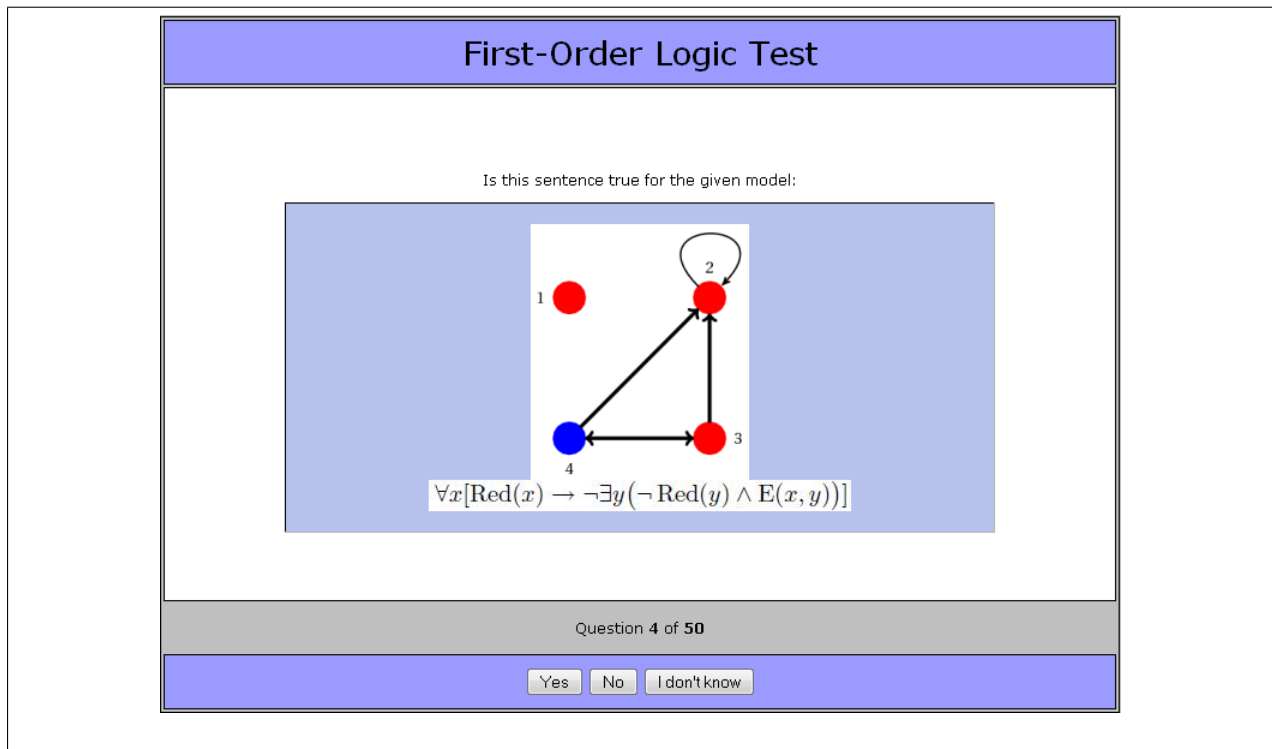
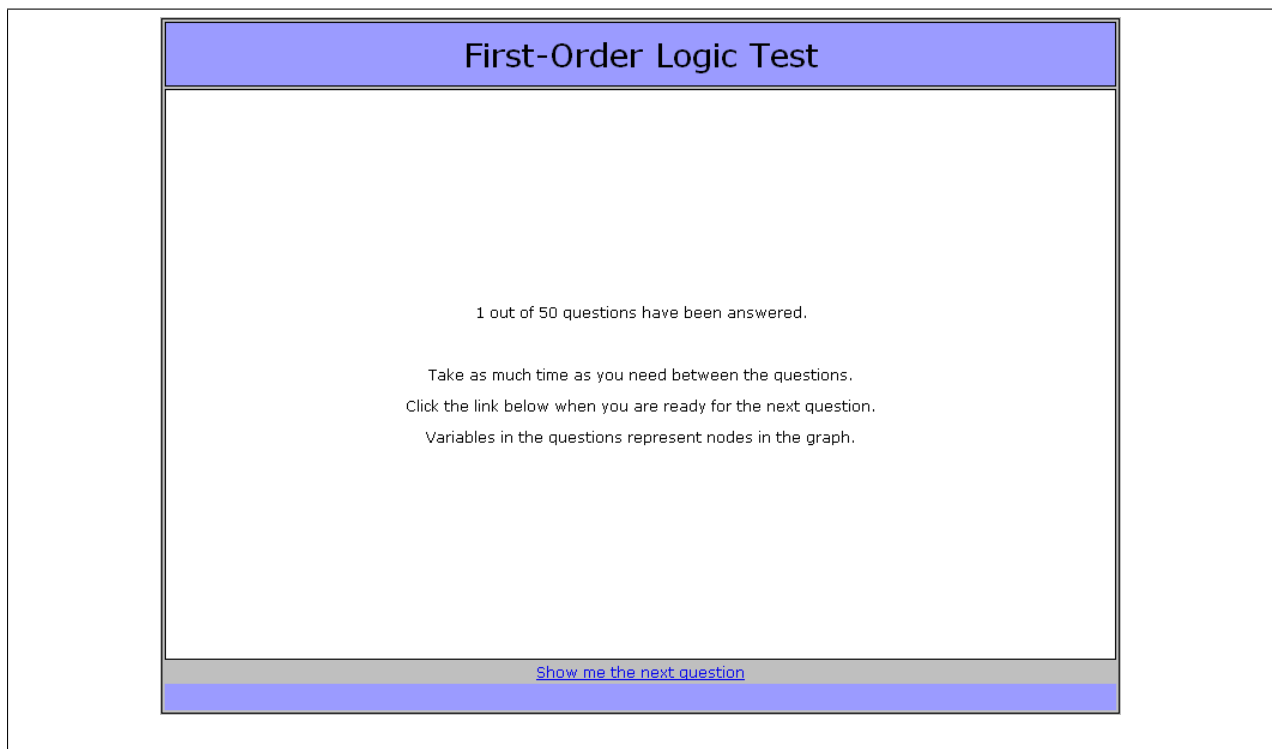
At the bottom of the form, there is a blue bar containing a button labeled "Start the test".

Figure 4.2: Web Interface: User Particulars

Following psychological measures of the trials were gathered from the experiment.

Accuracy This is defined as the number of correct answers for the test questions. Each question has an accuracy value of 0-10: the number of participants correctly classifying the truth value of a problem.

Latency This is the average response time for every question, for only the correct answers. Median values are used instead of mean to minimize the effects of possible unusual or extreme response times which may possibly occur due to external disturbances.

**Figure 4.3:** Web Interface: Trial Screen**Figure 4.4:** Web Interface: Next Question Screen

Chapter 5

Results

This chapter summarizes the results of the experiment conducted to study the psychological complexity of the FOL models and to compare it with the complexity of the proofs generated with the proposed formalism. Section 5.1 presents summarized results of the experiment with respect to accuracy and latency (response times). Regression graphs are presented for comparison of different complexity measures. Section 5.2.1 presents the results of the proofs and the usage of rules in the proofs. For discussion of these results, see the following chapter.

5.1 Results of the experiment

Ten participants attempted the test consisting of fifty problems, among them 24 true and 26 false questions. The complete results of the test are provided in Appendix C. Table 5.1 lists the results of the test for each participant for the 50 questions, including 24 true and 26 false trials.

Table 5.1: Results of the test

	True (24)	False (26)	Total (50)
Person 1	23 / 0 / 1	24 / 2 / 0	47 / 2 / 1
Person 2	14 / 9 / 1	17 / 9 / 0	31 / 18 / 1
Person 3	15 / 8 / 1	11 / 14 / 1	26 / 22 / 2
Person 4	23 / 1 / 0	21 / 5 / 0	44 / 6 / 0
Person 5	12 / 11 / 1	16 / 10 / 0	28 / 21 / 1
Person 6	20 / 3 / 1	24 / 1 / 1	44 / 4 / 2
Person 7	19 / 3 / 2	17 / 8 / 1	36 / 11 / 3
Person 8	16 / 7 / 1	16 / 10 / 0	32 / 17 / 1
Person 9	20 / 3 / 1	17 / 8 / 1	37 / 11 / 2
Person 10	16 / 7 / 1	11 / 14 / 1	27 / 21 / 2
Average	17.8 / 5.2 / 1.0	17.4 / 8.1 / 0.5	35.2 / 13.3 / 1.5
Percentage	74.1 / 21.7 / 4.2	66.9 / 31.2 / 1.9	70.4 / 26.6 / 3.0

5.1.1 Accuracy

Table 5.1 lists the accuracy data for the participants in the form of *correct answers / incorrect answers / timeouts*. For simplicity, all incorrectly answered and unanswered trials are considered as incorrect. Timeouts are also incorrect answers, but are listed separately to elaborate their proportion.

For individual problems, the accuracy value ranged from 2 to 10, with an average value of 7.

5.1.2 Latency

For the individual problems, the median response times for the correct answers were in the range 11.34 to 60.3, with the average value of 35.34.

5.2 Proofs

The proof finder was used to generate the minimum sized proofs for all the questions used in the test. The proofs were generated with minimum proof size, our main complexity measure. Proof lengths of the same proofs were also recorded and are used in the analysis for comparison only. This measure should not be confused with minimum proof length, for which the proof finder might have generated different proofs.

Due to the restriction of cognitive resources including working memory and visual memory, the proof finder was unable to generate proofs for three questions: 2, 5 and 45. Although the proof system is designed to be used for simple formulas, these questions are not really hard and were solved by the participants. This may be due to some problem with either the proof finder or the proof system itself. In any case, further improvement is required for both.

5.2.1 Usage of Rules

Table 5.2 summarizes the number of times each rule is used in the generated proofs for 47 questions, among them 22 true and 25 false ones.

5.3 Correlation

The results of the experiment were related to different complexity measures of the problems to see any relationship between them.

In the following table, latency is compared to the three proof measures and the correlation coefficients show that the proofs of the problems are correlated with it to some extent. The false problems have a higher correlation with latency.

Table 5.2: Usage of rules in the proofs

Rule Name	True	False	Total
Formula Inspect	66	63	129
Delta Inspect	22	31	53
Pi Inspect Prime	33		33
Weakening		24	24
Sigma Inspect Prime	2	26	28
Strengthening	23		23
Contradiction Recall		25	25
Tautology Recall	22		22
Pi Arrow Comprehension	2	5	7
Exists Reduce+	17	12	29
Rewrite	14	15	29
Forall Reduce+	9	9	18
Sigma And Comprehension	3	1	4

Table 5.3: Correlation of Proof Measures with Latency

Rule Name	True	False	All Problems
Proof Length	0.20	0.62	0.37
Min Proof Size	0.24	0.63	0.40
Proof Volume	0.36	0.75	0.52

The proof volume was computed by varying the weights of the rules to find the optimal ones. No special method was applied for this due to the high computing requirements, and the weights were found manually. Therefore, the proof volume was not used later on for further analysis and was abandoned at this stage.

The complexity measures are not related to accuracy values in the test, showing that the proof rules actually model the latency instead of accuracy.

Table 5.4: Correlation of Proof Measures with Accuracy

Rule Name	True	False	All Problems
Proof Length	-0.16	-0.02	-0.04
Min Proof Size	-0.20	-0.03	-0.06
Proof Volume	-0.36	0.06	-0.08

The figures 5.1, 5.2 and 5.3 show the correlation of various mathematical measures with latency. The proof measures show a better relationship with latency than the formula length except a few points which are scattered away from the trend line.

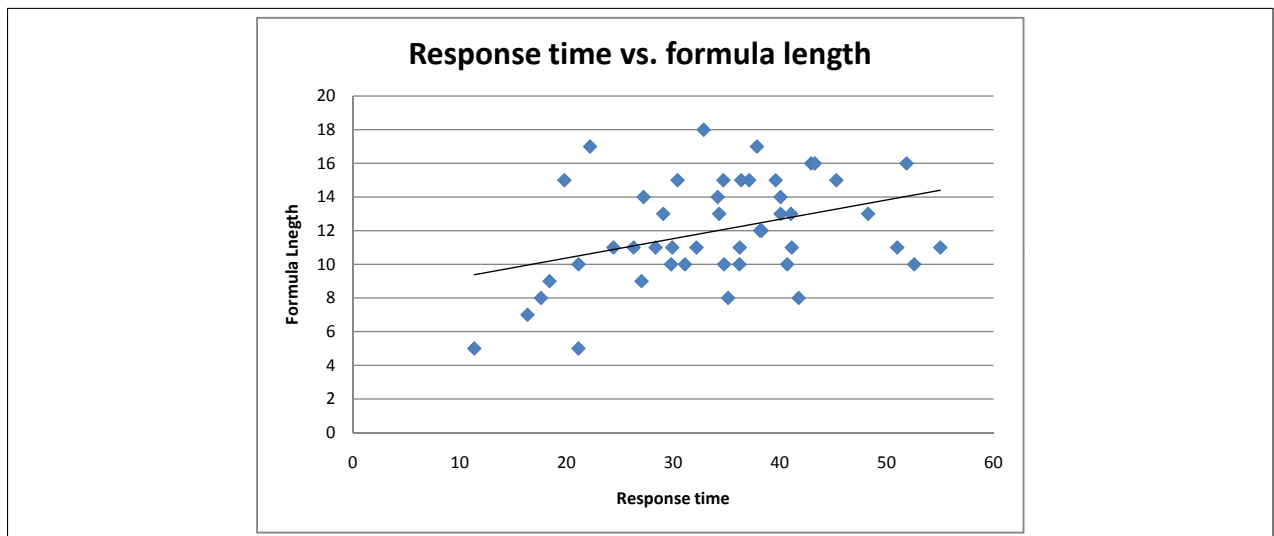


Figure 5.1: Formula Length vs. Latency

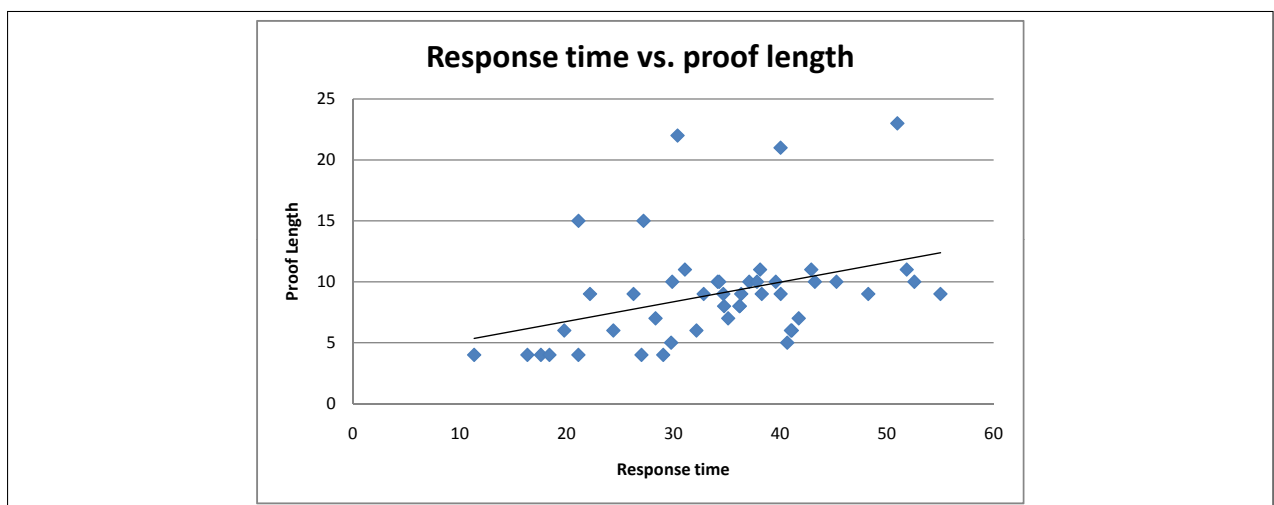


Figure 5.2: Proof Length vs. Latency

This shows that the proof measures model the psychological complexity of the problems better than the simple measures such as formula length. However, some of the proofs are oversmart in this system, suggesting that more work needs to be done to improve the proof model.

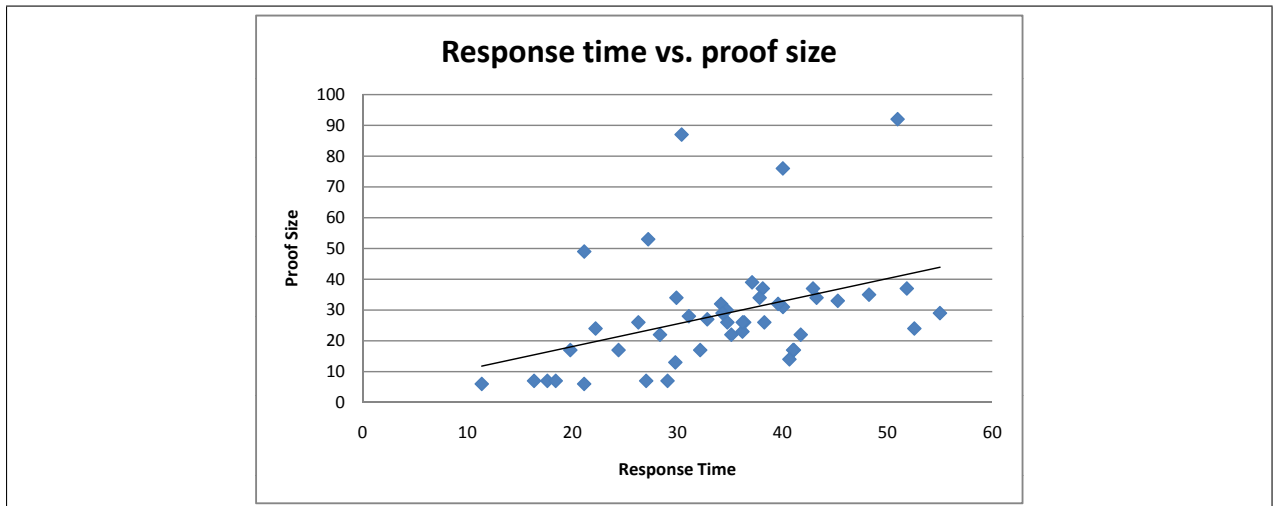


Figure 5.3: Proof Size vs. Latency

Chapter 6

Discussion

6.1 Interviews with Participants

After the test, some short interviews were done with the participants to see what they thought about the test and how did they solve the problems. The interviews showed that understanding the formula was a major step consuming more than 50% of the time. On the other hand, understanding the model was trivial and solving the problem did not take that much once they had analyzed the formula. The problems were presented with both the formula and the graph appearing simultaneously, but most participants said they looked at the formula first and tried to analyze it. They also revealed that although some tried to analyze the complete formula before beginning to solve the problem, some of them indeed tried to look at the parts if the formula was long. Most of them told that they used the strategy to falsify the formula first by trying to find a counter example, and if failed, analyzing it for the truth.

It was earlier suggested by Lance Rips that the underlying axioms must be conveyed clearly to the participants to avoid misunderstanding. He emphasized that this was most important, and that any misconception will lead to false results. This proved to be true, as a few of the subjects had really a problem in understanding predicate $E(x, y)$. They were in doubt whether x and y in this term can be instantiated to the same node or not. This was made clear to them during the break that multiple variables can be instantiated to the same node. But even being in doubt, they mostly evaluated it in the right way as the use of self-loops in most models was a guiding factor to them.

6.2 Formula and Model Complexities

As pointed out by some participants, the amount of time taken by them to understand the formula was considerably higher as compared to the total time spent to solve a particular problem. This is also evident in the correlations between model measures and test measures. Table 6.1 gives such correlations between the different variables of the two types, for the subset of 47 questions for which the proof generator was able to generate proofs. The variables are explained as follows:

Nodes : Number of nodes used in the model

Edges : Number of edges (uni- or bidirectional) in the model

Cons : Number of connections (directed edges) in the model

Colors : Number of different colors used in the model

Table 6.1: Correlation of Model Complexities

Model Measures	Latency	Accuracy
Nodes	0.072	0.058
Edges	-0.112	-0.012
Cons	-0.169	-0.039
Colors	-0.167	0.167

This suggests that the model complexity has little or no effect on the human reasoning for solving these questions, whereas the formula complexity plays a bigger role in the same. This may be due to the fact that the model size was kept limited to an average 3.5 nodes per model (3-4 nodes in each), and 3.76 edges per model (3-5 in each). Further, the fact that graphs are visually appealing and do not need analytical digestion, is also an important factor in this. Whereas the logic formulas are quite complex for human reasoning, which is evident from table 6.2 listing the correlations of formula measures with latency and accuracy.

Table 6.2: Correlation of Formula Complexities

Formula Measures	Latency	Accuracy
Length	0.374	-0.138
Quantifiers	0.525	-0.138
Binary Connectives	0.231	-0.023
Parse Depth	0.496	-0.292
Predicates	0.231	-0.023
Negations	0.197	-0.376

6.3 Multiple Correlation

”The general purpose of multiple regression (the term was first used by Pearson, 1908) is to learn more about the relationship between several independent or predictor variables and a dependent or criterion variable.” [16]

During the course of evaluation of results, multiple correlation was applied to see any significant relationship between the problem measures and test results.

6.3.1 Model Measures

All the four model measures (number of nodes, edges, colors and connections) were correlated with the latency and accuracy. The results, in accordance with our previous

observation, show that the model measures do not significantly affect the test results. The correlation coefficient was 0.068 for latency and 0.158 for accuracy.

6.3.2 Formula Measures

The combined regression of all the formula measures was 0.60 with latency and 0.43 with accuracy. This shows that the formula complexity affects the human performance in the test to a large extent, and can be used to predict the psychological complexity, though with less precision.

Considering that the problems comprised of both true and false questions, truth value was introduced as another formula measure, being 0 for false questions and 1 for true questions. When this was included in the set of formula measures for multiple correlation, the result was a correlation of 0.67 with latency and 0.55 for accuracy, which is somewhat better than the previous values.

However, when the proof measures were included in the set of predictors, no considerable achievement was observed. The two measures included were the proof length and the proof size. The new correlation was 0.70 with latency (compared to 0.67 without proof measures) and 0.55 with accuracy (compared to the previous 0.54), which does not improve the previous values.

6.4 Selective Regression

In order to further analyze the results, regression was applied on subsets of values selected on various criteria.

6.4.1 Selecting by Higher accuracy

The participants were not equally good at solving the test problems. On the other axis, not all questions were solved by a good number of participants, with some problems solved only by less than half of them. Selective regression was applied on the measures of problems with higher accuracy. First we looked at the accuracy distribution of the test problems, given in table 6.3.

As the table 6.3 shows, exactly half of the problems were solved with a very high accuracy of 80% and higher. This was selected as the threshold and those problems with accuracy 80% and higher were selected for analysis. Out of those 25, we did not have proof measures for two of them, making the total to 23 problems. The results of regression for this selected subset of problems are given in table 6.4.

The proof size, our main mathematical complexity measure, is highly correlated with latency for the false questions. The values are better than the correlations in the full data, but with less data points. Correlation coefficients for formula length and proof length are provided for comparison. Values for formula length show that it is still a better predictor for the latency than the proof measures. Higher coefficients for false problems show that the system reasonably models those problems but fails to properly model the true questions.

Table 6.3: Accuracy distribution of test problems

Accuracy	No. of Problems	%
1	0	0%
2	1	2%
3	0	0%
4	6	12%
5	2	4%
6	7	14%
7	9	18%
8	17	34%
9	5	10%
10	3	6%
Tot	50	100%

Table 6.4: Selective Regression Analysis - 80% and higher

	True	False	Combined
Formula length	0.38	0.90	0.54
Proof length	0.42	0.70	0.43
Proof size	0.47	0.85	0.51

In comparison, the regression of the questions with accuracy less than 80% is listed in table 6.5, for 24 questions, among them 9 true and 15 false. The correlation of formula length with latency is visibly much less than the above values, suggesting that the formula length is better correlated to latency in easier problems. Values for proof measures are also less than those in the above table.

Table 6.5: Selective Regression Analysis - less than 80%

	True	False	Combined
Formula length	-0.48	0.31	0.07
Proof length	-0.39	0.58	0.25
Proof size	-0.31	0.55	0.24

6.4.2 Selecting by Formula Complexity

Correlations of mathematical measures with the latency show that the false problems are better correlated than the true ones. This leads us to think why is it so, when the system has almost similar rules for both types of problems. The formulas were analyzed for any

sign related to this, and it was observed that some of the formulas might be harder to solve than others.

From the 24 true problems, 8 were universal formulas (starting with \forall) and the rest were existential (starting with \exists). To solve a true existential formula, one only needs to find one example to prove it true. Whereas, the truth of universal formulas has to be checked in the whole domain, making it harder to solve.

Looking at the false problems, only two were found as existential formulas which may be harder to solve than the false universal formulas. This clue was pointed out by one of the team members. A larger number of universal true formulas as compared to existential false formulas may be the cause of the bad correlations of true problems earlier. From here on, I will refer to these 10 problems as harder formulas and for the rest, including true existential and false universal, as easier formulas.

For analysis, the harder formulas (1,5,11,13,15,16,17,19,42,45) were removed from the dataset and the rest were analyzed, except problem 2 for which no proof measures were available. For these 39 problems, the correlation of 16 true ones stands at 0.65, for the 23 false ones it stands at 0.69, combined the correlation coefficient is 0.66. Unlike previous figures, the current values do not have a wide difference for true and false problems.

This important finding shows that the proof system better models the easier formulas, but not the harder ones as pointed out above. This finding is also supported by the participant interviews where many of them said they always tried to find a counter-example first to solve the problem.

On the other hand, the harder formulas had a correlation of only 0.28 in latency vs. proof size (for only 8 of those 10, as problems 5 and 45 did not have proof measures).

If we combine the two selection criteria discussed above, i.e. select problems with accuracy 80% and higher, and then select the easier problems from those, we get a set of 19 problems. The correlation of proof size of this selection with the latency stands at 0.76 which is promising, as suggested by Lance Rips that a correlation of at least 0.70 to 0.80 is sufficient to prove that the system actually models human reasoning in first order logic. Although it comes at the cost of reduced data set, which shows the proof system does need further refinement and improvement. For comparison, the formula length vs. latency correlation for this subset is 0.52.

Chapter 7

Conclusion

7.1 Summary and Conclusion

The main conclusion drawn from this thesis is that better proof systems can be defined for classical logic than the traditional formalisms, which can incorporate concepts from cognitive psychology and model the actual human reasoning in logic. It can be deduced from the results of the experiment that the psychological complexity of a problem in first order logic can be predicted by the suggested model, though it needs more development. The minimum proof size of the proofs in the suggested model highly correlates to the average response times of high accuracy questions in the experiment. The suggested model can be developed further by using different axioms and rules, using better complexity measures, conducting more psychological experiments, and in several other ways.

7.2 Future Work

The work presented in this thesis is just a startup to a broader field of applying cognitive science concepts in the computational problem solving. The scope of this study was limited to first order logic. The natural next step to this study will be to further the development of the proof strategy presented herein and to make it closer to the human reasoning. One of the important developments will be to verify the findings of section 6.4.2 and develop the rules for the harder questions as discussed therein.

The memory model used in this study is a basic model of human memory, whereas the models currently used in cognitive science are quite complex. To better assess the human reasoning, a more sophisticated model can be implemented which will hopefully yield better results.

Bibliography

- [1] Strannegård, C.: Anthropomorphic Artificial Intelligence. Kapten Mnemos kolumbarium, Department of Philosophy, Göteborg University (2005)
- [2] Miller, G.A.: The magical number 7, plus or minus two: Some limits on our capacity for processing information. *The Psychological Review*, vol. 63 (1956)
- [3] Ram, H., Freed, K.: Finding patterns in series: Measuring complexity of human recollection. Department of Computer Science and Engineering, Göteborg University (2006)
- [4] Strannegård, C.: Proving first-order sentences with bounded cognitive resources. *Philosophical Communications, Web Services*, No 39, Department of Philosophy, Göteborg University (2007)
- [5] Strannegård, C., et.al.: Reasoning Processes in Propositional Logic. *Journal of Logic, Language and Information*, vol. 19 issue 3: 283-314 (2010)
- [6] Smith, R.E., Passer, M.W.: *Psychology: The Science of Mind and Behavior*. McGraw-Hill (2008)
- [7] Gentzen, G.: Investigations into logical deductions. In: Szabo, M.E. (ed.): *The Collected Papers of Gerhard Gentzen*. North-Holland Publishing Co., Amsterdam (1969)
- [8] Jaśkowski, S.: On the rules of suppositions in formal logic. *Studia Logica* 1, 5-32 (1934). Reprinted in: S. McCall (ed.), *Polish Logic 1920-1939*, Clarendon Press, Oxford, pp. 232-258.
- [9] Claes Strannegård, Kristofer Sundén Ringnér, John Hughes: *A Case Study in Anthropomorphic AI*. (2005)
- [10] Smith, R.E., Passer, M.W.: *Psychology: The Science of Mind and Behavior*. McGraw-Hill (2008)
- [11] Atkinson, R.C., Shiffrin, R.M.: Chapter: Human memory: A proposed system and its control processes. In Spence, K.W., Spence, J.T.: *The psychology of learning and motivation (Volume 2)*. New York: Academic Press. pp. 89-195 (1968)
- [12] Baddeley, A.: *Working Memory, thought and action*. Oxford University Press (2007)
- [13] Braine, M.D.S., O'Brien, D.P.: *Mental logic*. L. Erlbaum Associates (1998)

- [14] Newstead, S.: Interpretational errors in syllogistic reasoning. *Journal of Memory and Language* 28, 78-91 (1989)
- [15] Newstead, S.: Gricean implicatures and syllogistic reasoning. *Journal of Memory and Language* 34, 644-664 (1995)
- [16] Multiple Regression. In: *Electronic Statistics Textbook*, StatSoft, Inc. (2010), Tulsa, OK, Web: www.statsoft.com/textbook/multiple-regression, accessed 2010-08-10.
- [17] Hedqvist, D.: Human reasoning in propositional logic. Master's thesis, Chalmers University of Technology (2007)

Appendix A

List of Test questions

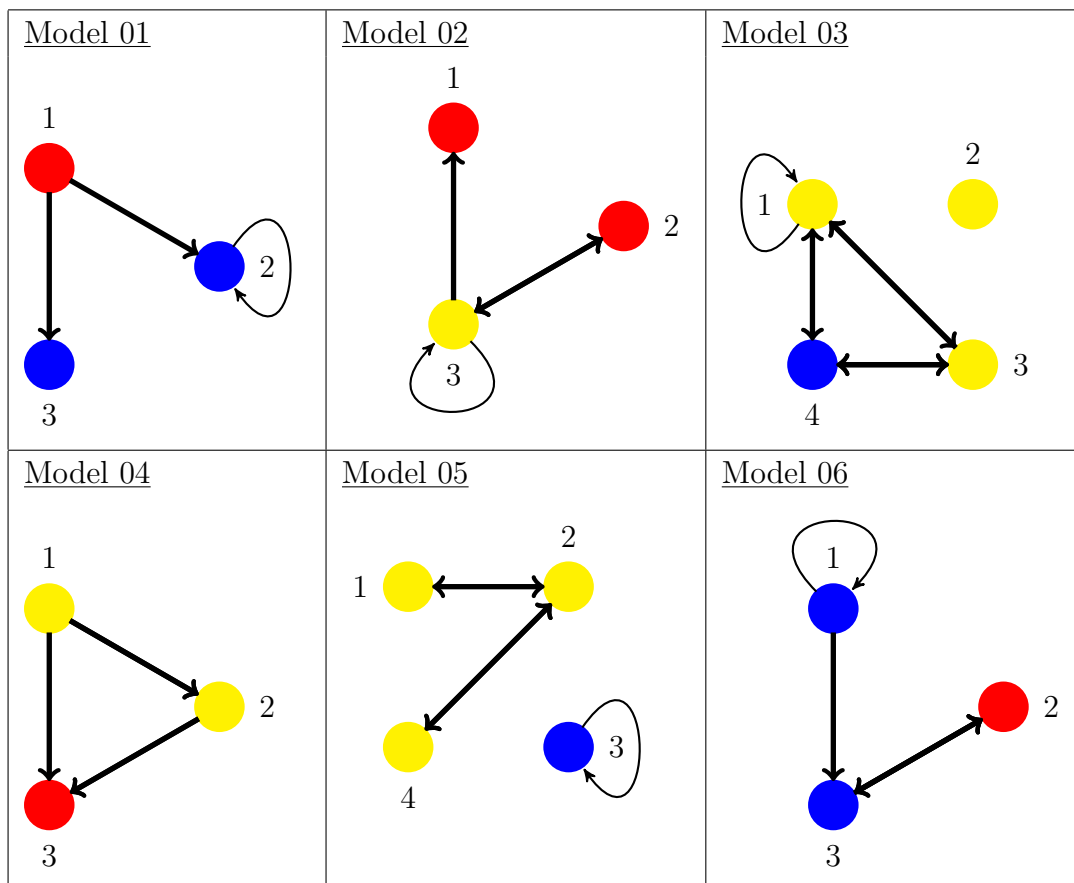
The fol formulas used in the experiment are listed here along with their truth value in the given model (models listed in Appendix B). Proof values for questions 2, 5 and 45 are not given, as these could not be computed within the specified limit of cognitive resources. Here FL = formula length, PL = proof length, and PS = minimum proof size.

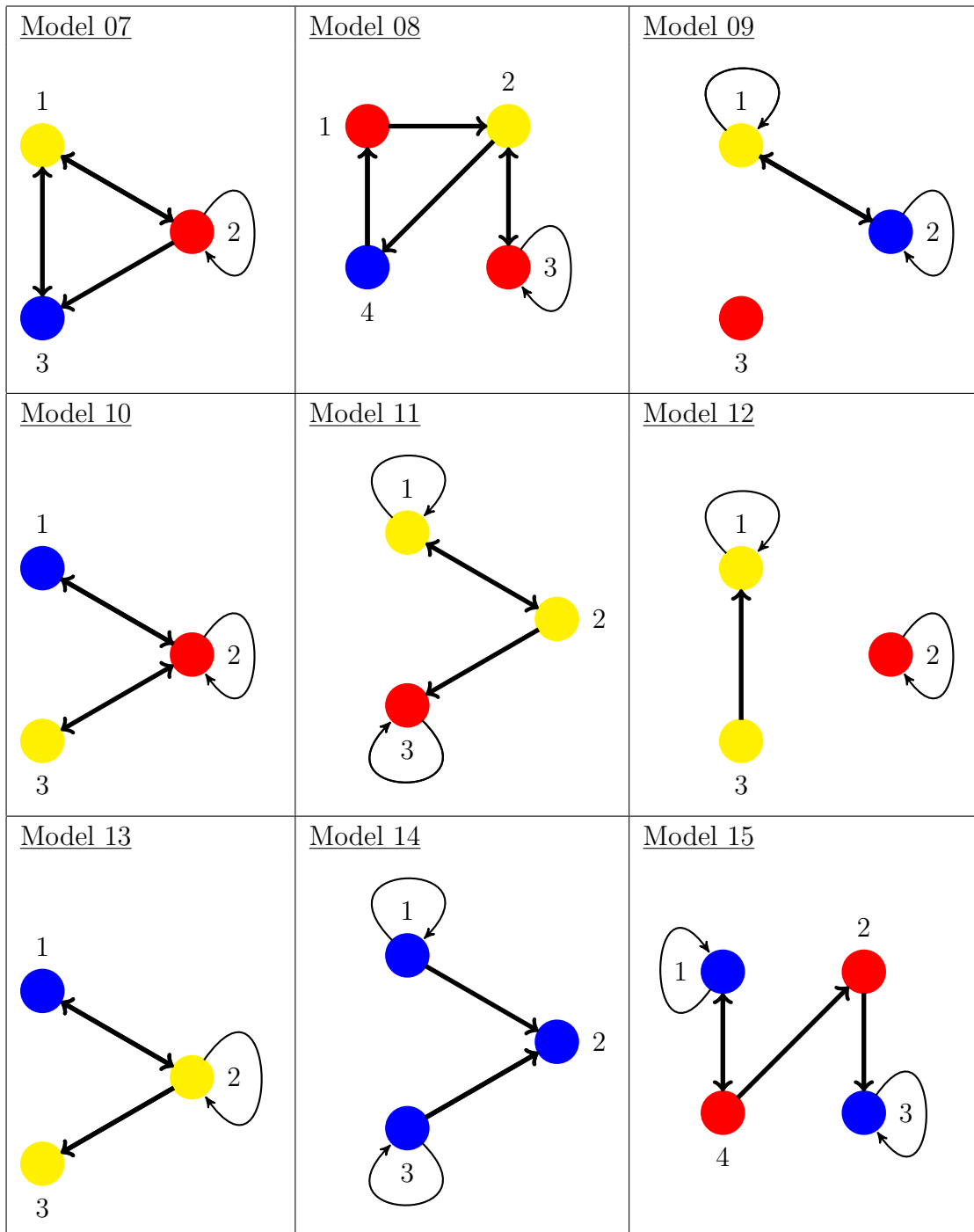
#	Formula	Truth	FL	PL	PS
1	$\exists x(\text{Blue}(x) \wedge \forall y[\text{Blue}(y) \rightarrow \text{E}(x, y)])$	False	13	9	35
2	$\forall x(\text{Red}(x) \rightarrow \exists y\exists z[\text{E}(x, y) \wedge \text{E}(y, z) \wedge \text{Red}(z)])$	False	19		
3	$\exists x[\text{Yellow}(x) \wedge \forall y\neg\text{E}(x, y)]$	True	11	10	34
4	$\forall x\forall y[\neg\text{E}(x, y) \rightarrow \text{Yellow}(y)]$	False	11	6	17
5	$\forall x(\exists y[\text{Yellow}(y) \wedge \text{E}(x, y)] \leftrightarrow \text{Yellow}(x))$	True	13		
6	$\forall x([\text{E}(x, x) \vee \neg\text{Blue}(x)] \rightarrow \text{E}(3, x))$	False	13	4	7
7	$\exists x\exists y[\text{Red}(x) \wedge \text{Red}(y) \wedge \text{E}(x, y) \wedge \text{E}(y, x)]$	True	17	9	24
8	$\exists x\exists y\exists z[\text{E}(x, y) \wedge \text{E}(y, z) \wedge \neg\text{E}(z, x)]$	True	18	9	27
9	$\forall x[\neg\text{Red}(x) \rightarrow \forall y\text{E}(x, y)]$	False	11	9	26
10	$\forall x\forall y[\text{E}(x, y) \wedge \text{E}(y, x)]$	False	11	6	17
11	$\forall x[\neg\text{E}(x, x) \rightarrow \neg\text{Red}(x)]$	True	10	5	13
12	$\forall x(\text{E}(x, x) \rightarrow \exists y[\text{E}(x, y) \wedge \text{Yellow}(y)])$	False	14	10	32
13	$\forall x(\neg\text{E}(x, x) \rightarrow \exists y[\text{Yellow}(y) \wedge \text{E}(y, x)])$	True	15	10	39
14	$\exists x\forall y\neg\text{E}(x, y)$	True	8	7	22
15	$\forall x(\neg\text{E}(x, x) \rightarrow \exists y[\text{Blue}(y) \wedge \text{E}(x, y)])$	True	15	22	87
16	$\exists x\neg\exists y[\text{Blue}(y) \wedge \text{E}(y, x)]$	False	11	23	92
17	$\forall x(\text{Red}(x) \rightarrow \exists y[\text{E}(x, y) \wedge \neg\text{Blue}(y)])$	True	14	15	53
18	$\forall x[\text{Red}(x) \rightarrow \text{E}(x, x)]$	False	8	4	7

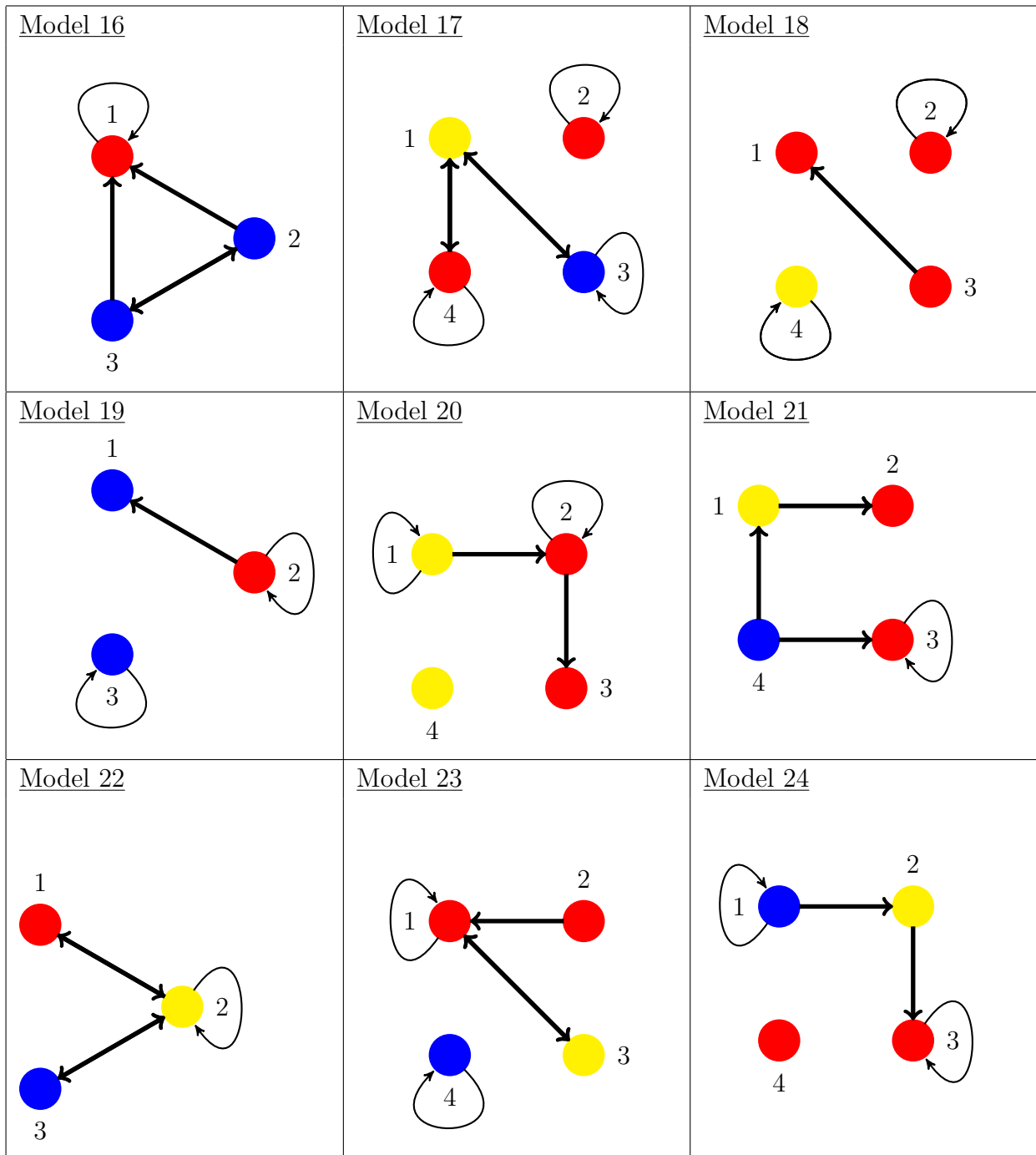
#	Formula	Truth	FL	PL	PS
19	$\forall x(E(x, x) \rightarrow \exists y[\text{Blue}(y) \wedge E(x, y)])$	True	14	21	76
20	$\exists x[\neg \text{Yellow}(x) \wedge \forall y\neg E(x, y)]$	True	12	11	37
21	$\forall x\exists y[E(x, y) \wedge \text{Red}(y)]$	False	10	8	26
22	$\exists x\forall y[E(x, y) \wedge E(y, x)]$	True	11	7	22
23	$\exists x[\neg E(x, x) \wedge \exists yE(y, x)]$	True	12	9	26
24	$\forall x\forall y[E(x, y) \leftrightarrow E(x, x)]$	False	11	6	17
25	$\forall x(\text{Yellow}(x) \leftrightarrow \exists y[\text{Yellow}(y) \wedge E(x, y)])$	False	13	9	31
26	$\forall x\forall y([\text{Blue}(x) \wedge \text{Yellow}(y)] \rightarrow E(x, y))$	False	13	6	17
27	$\forall x(\neg E(x, x) \rightarrow \exists y[E(x, y) \wedge \neg \text{Blue}(y)])$	False	16	11	37
28	$\exists x(\text{Yellow}(x) \wedge \neg \exists y[\neg \text{Yellow}(y) \wedge E(x, y)])$	True	15	10	33
29	$\exists x\exists y[E(x, x) \wedge E(y, y) \wedge E(x, y)]$	True	15	6	17
30	$\exists x[\forall yE(x, y) \rightarrow \text{Blue}(x)]$	True	10	5	14
31	$\forall x[\text{Red}(x) \rightarrow \neg \exists y(\neg \text{Red}(y) \wedge E(x, y))]$	False	15	9	30
32	$\forall x[\text{Yellow}(x) \rightarrow \forall yE(x, y)]$	False	10	8	23
33	$\forall x(E(x, 2) \vee \neg \exists y[\text{Blue}(y) \wedge E(y, x)])$	False	15	10	32
34	$\forall xE(x, 3)$	False	5	4	6
35	$\forall x[E(x, 2) \wedge E(x, 3)]$	False	9	4	7
36	$\forall x(\text{Yellow}(x) \rightarrow \exists y[\text{Red}(y) \wedge E(x, y)])$	False	13	10	29
37	$\exists x\exists y[\neg \text{Yellow}(x) \wedge \text{Yellow}(y) \wedge \neg E(x, y)]$	True	15	9	26
38	$\forall x[\exists yE(x, y) \rightarrow E(x, x)]$	False	11	9	29
39	$\forall x[\neg E(x, x) \rightarrow \exists y(E(y, y) \wedge E(x, y))]$	False	16	11	37
40	$\forall x[\forall yE(x, y) \leftrightarrow \text{Blue}(x)]$	False	10	10	24
41	$\exists x[\text{Blue}(x) \wedge \forall y(\text{Yellow}(y) \rightarrow [E(x, y) \vee E(y, x)])]$	True	17	10	34
42	$\forall x[\text{Red}(x) \rightarrow \exists yE(x, y)]$	True	10	15	49
43	$\forall x[\text{Red}(x) \vee \text{Yellow}(x)]$	False	7	4	7
44	$\forall x[\text{Yellow}(x) \rightarrow \exists yE(x, y)]$	False	10	11	28
45	$\forall x\forall y([E(x, y) \wedge E(y, x)] \rightarrow [E(x, x) \vee E(y, y)])$	True	19		
46	$\exists x\forall y[\text{Blue}(y) \rightarrow \neg E(x, y)]$	True	11	8	26
47	$\forall x[\text{Yellow}(x) \rightarrow \exists y(E(x, y) \wedge (\text{Blue}(y) \vee \text{Red}(y)))]$	False	16	10	34
48	$\exists xE(x, x)$	True	5	4	6
49	$\exists x\forall y\neg E(x, y)$	True	8	7	22
50	$\exists x[\neg \text{Blue}(x) \wedge E(x, x)]$	True	9	4	7

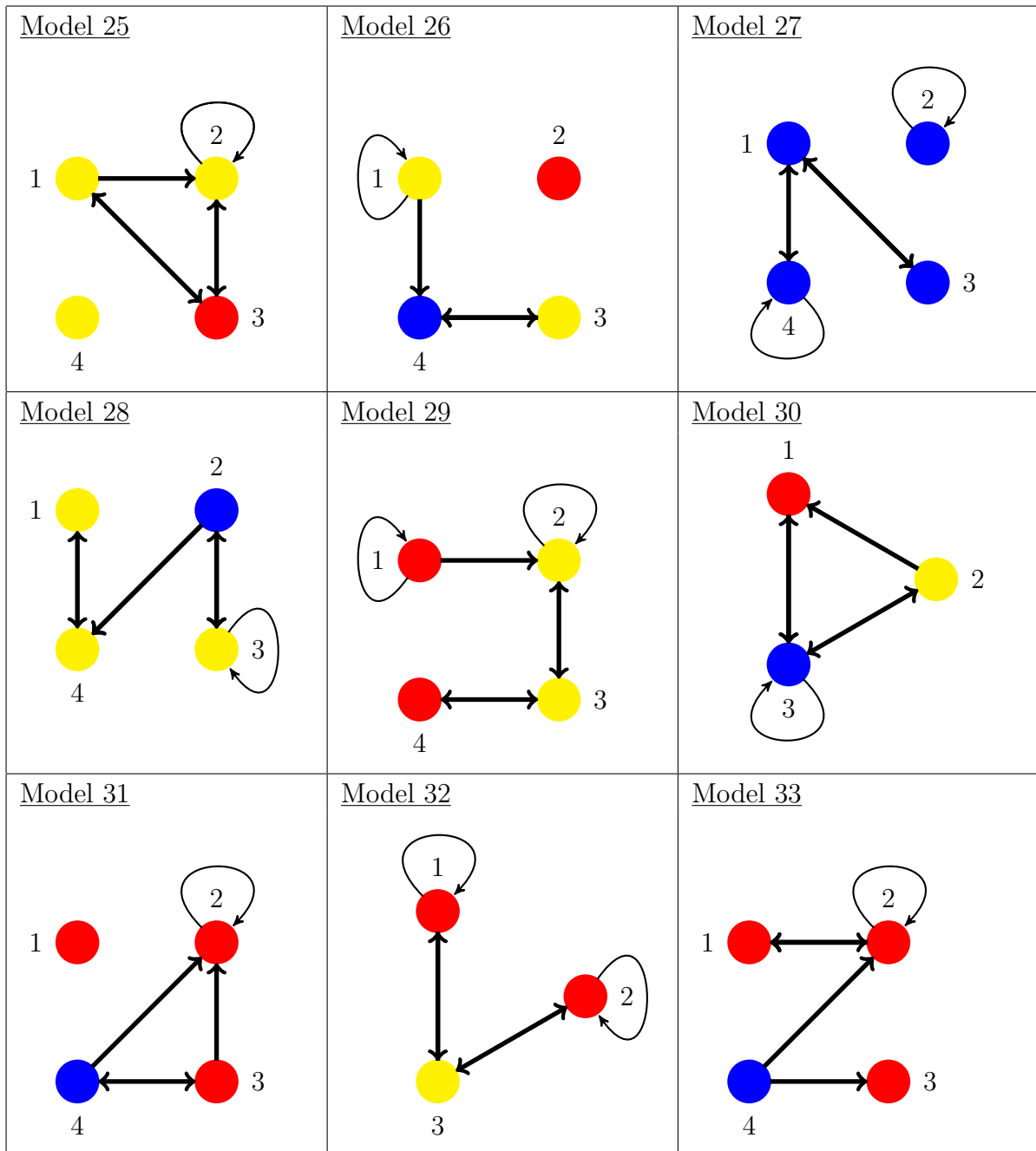
Appendix B

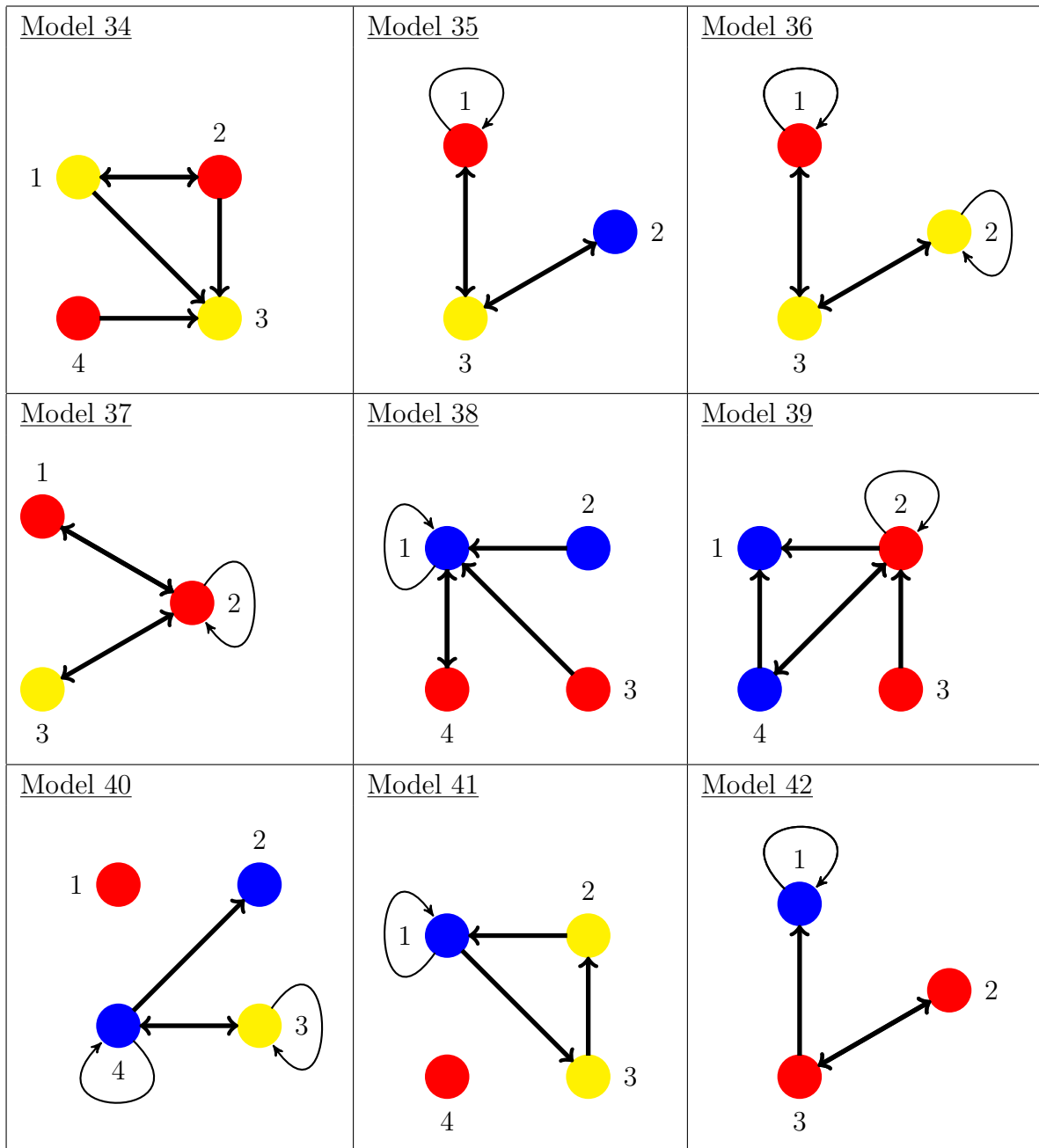
Models for the Test Questions

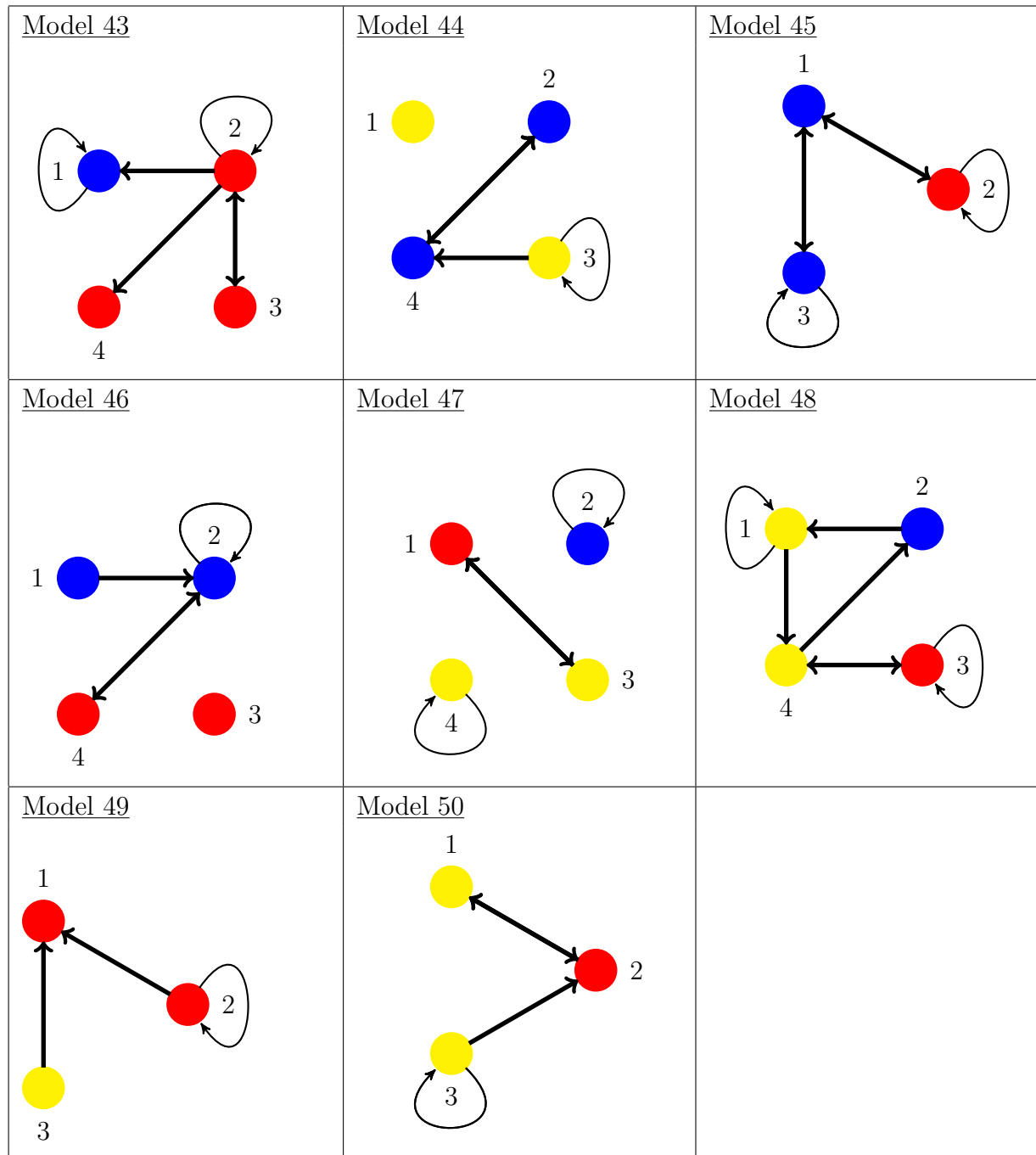












Appendix C

Test Results

The following table lists the response times of the 10 participants for the test questions. Only correct answers are listed here.

The following parameters are provided at the end of the table.

- avg = Average
- tot = Total
- rel = Reliability

Reliability is a parameter to assess the performance of users, and is defined as 'one minus the ratio of wrong answers'.

APPENDIX C. TEST RESULTS

#	P-1	P-2	P-3	P-4	P-5	P-6	P-7	P-8	P-9	P-10	Median	Accuracy
1	47.231	32.377		26.926		49.287	57.578	72.735	10.719	66.640	48.259	8
2	30.264	67.878		35.266	42.565	65.716	54.906		70.516	70.484	60.311	8
3	17.031	23.173	21.172	12.422	47.128		59.984	29.906	30.704	45.422	29.906	9
4				14.025				68.172			41.098	2
5	52.169			31.422			24.546	60.907	71.047	53.297	52.733	6
6	29.061					49.344	73.547	21.500	20.219		29.061	5
7	19.202	29.783		22.865	18.563	19.315	22.203		24.407		22.203	7
8	39.639		25.749	16.435	58.800	37.502	28.188	73.344	26.047		32.845	8
9	21.046	31.517		14.670					48.516		26.281	4
10	19.250				24.392	29.054	27.219		23.469		24.392	5
11	30.545	61.128		6.279		30.012	15.781	29.594	10.015	38.437	29.803	8
12	23.749	65.785		14.936	37.877	34.163	18.344		43.734		34.163	7
13	32.686		41.546	10.047		51.584	16.766		46.172		37.116	6
14	24.749	50.237	41.686	13.078		21.877	49.640	41.813	42.657		41.749	8
15	28.031			27.777		32.585		28.218	62.563	48.016	30.401	6
16	45.497	68.800	52.734	25.535		49.230	84.765				50.982	6
17	23.640		23.234	23.749		27.217	30.219	32.281	30.063		27.217	7
18	9.203	14.923		8.156	24.892	10.191	17.609	19.812	30.297	21.969	17.609	9
19	20.155	53.972	68.389	21.341	46.909	40.048	34.515	22.922	67.469		40.048	9
20	29.842			23.984	48.081	48.343			42.094	34.172	38.133	6
21	39.015		37.015	17.406	28.877	19.691	49.687	32.500	47.578		34.757	8
22	23.687	38.424	62.889	13.562		28.341	24.562	45.062	61.016	13.578	28.341	9
23	30.171	48.815	28.594	24.559	38.284	41.325	26.937		50.438	46.890	38.284	9
24	33.202	39.190	36.656	21.559		22.741	31.141	68.093		28.187	32.171	8

#	P-1	P-2	P-3	P-4	P-5	P-6	P-7	P-8	P-9	P-10	Median	Accuracy
25	19.593	53.581	58.889	17.094	37.034	19.191	43.078	48.546			40.056	8
26	23.202	40.268	41.780	23.984	70.363	54.942		38.516	59.891		41.024	8
27	27.482		42.921	22.999		32.460	64.953		42.969	57.562	42.921	7
28	48.435			19.328		74.762				42.125	45.280	4
29	15.795	30.283	22.577	11.385	17.016	31.361		10.891		34.219	19.796	8
30		36.924		42.219	70.801		30.156	54.562	40.672	21.109	40.672	7
31	29.733		70.202	33.511		35.865					34.688	4
32					56.675	34.912		18.829	37.484		36.198	4
33	61.246	40.752		38.437		33.772					39.594	4
34	7.406		24.999		13.532	21.136		21.094	23.140		21.115	6
35	27.514	58.878		20.593	26.533	30.038	18.235		19.875	47.172	27.023	8
36	19.686	53.769		13.369	36.612	26.507		34.297		65.781	34.297	7
37	30.733			13.010	26.736	36.364	51.000		38.328	55.453	36.364	7
38	23.578	57.175		34.186		64.032	55.015	33.922		61.719	55.015	7
39	51.356	75.145			52.363	38.926	61.937	24.672			51.859	6
40	52.575		32.234	66.522	41.190	52.910		23.531	66.016		52.575	7
41	25.389	53.456	43.390	28.294		26.993	34.484	41.187		65.797	37.835	8
42	18.202		22.750	16.437		33.064	19.499	17.203	63.688	32.562	21.124	8
43	8.547	23.142		6.094	16.641	16.579	16.110		11.656	49.469	16.344	8
44	13.453	25.861	32.500	11.558	32.596	29.687	51.343	14.016	35.516	38.625	31.093	10
45	33.639	51.143	74.576	19.341		36.906	48.375	53.219	52.516		49.759	8
46	33.763			21.725			42.047		38.688		36.225	4
47	32.560	39.940	71.764	24.656	39.643	47.039	35.265	65.704	46.578	63.781	43.259	10
48	7.172	17.485	10.797	4.749	9.235	25.633		11.922		36.500	11.359	8

#	P-1	P-2	P-3	P-4	P-5	P-6	P-7	P-8	P-9	P-10	Median	Accuracy
49	25.811	33.314	34.686		54.534	46.108	42.687		35.594	29.750	35.140	8
50	9.313	22.595	20.140	4.811	12.814	20.258	22.860	16.672	9.937	53.719	18.406	10
avg	27.984	43.217	40.149	20.916	36.810	35.841	38.477	36.739	40.062	45.275		
tot	47	31	26	44	28	44	36	32	37	27		
rel	0.96	0.68	0.62	0.88	0.64	0.92	0.80	0.68	0.78	0.72		

Appendix D

Axioms

This appendix lists the axioms used in the declarative memory (DM). These are common formations and are supposed to be known by the test participants.

A, B and C denote arbitrary formulas. Most of the following were used in the earlier work presented in [5]. Some new additions include those for predicates and quantifiers.

D.1 Tautologies

Truth-table entries

\top
 $\neg\perp$
 $\top \wedge \top$

Identity

$A \vee \top$
 $\top \vee A$
 $A \rightarrow \top$
 $\perp \rightarrow A$
 $(A \vee \perp) \leftrightarrow A$
 $(\perp \vee A) \leftrightarrow A$
 $(A \wedge \top) \leftrightarrow A$
 $(\top \wedge A) \leftrightarrow A$
 $(\top \rightarrow A) \leftrightarrow A$
 $(A \rightarrow \perp) \leftrightarrow \neg A$
 $(A \leftrightarrow \top) \leftrightarrow A$
 $(\top \leftrightarrow A) \leftrightarrow A$
 $(\perp \leftrightarrow A) \leftrightarrow \neg A$
 $(A \leftrightarrow \perp) \leftrightarrow \neg A$

Idempotence

$$(A \vee A) \leftrightarrow A$$

$$(A \wedge A) \leftrightarrow A$$

$$A \rightarrow A$$

$$A \leftrightarrow A$$

Double Negation

$$\neg\neg\top$$

$$\neg\neg A \leftrightarrow A$$

Commutativity

$$(A \wedge B) \leftrightarrow (B \wedge A)$$

$$(A \vee B) \leftrightarrow (B \vee A)$$

$$(A \leftrightarrow B) \leftrightarrow (B \leftrightarrow A)$$

Associativity

$$(A \wedge B) \wedge C \leftrightarrow A \wedge (B \wedge C)$$

$$A \wedge (B \wedge C) \leftrightarrow (A \wedge B) \wedge C$$

$$(A \vee B) \vee C \leftrightarrow A \vee (B \vee C)$$

$$A \vee (B \vee C) \leftrightarrow (A \vee B) \vee C$$

$$((A \leftrightarrow B) \leftrightarrow C) \leftrightarrow (A \leftrightarrow (B \leftrightarrow C))$$

$$(A \leftrightarrow (B \leftrightarrow C)) \leftrightarrow ((A \leftrightarrow B) \leftrightarrow C)$$

Distributivity

$$(A \wedge B) \vee (A \wedge C) \leftrightarrow A \wedge (B \vee C)$$

$$(A \vee B) \wedge (A \vee C) \leftrightarrow A \vee (B \wedge C)$$

De Morgan

$$(\neg A \wedge \neg B) \leftrightarrow \neg(A \vee B)$$

$$\neg(\neg A \wedge \neg B) \leftrightarrow (A \vee B)$$

$$\neg(A \wedge \neg B) \leftrightarrow (\neg A \vee B)$$

$$\neg(\neg A \wedge B) \leftrightarrow (A \vee \neg B)$$

$$(\neg A \vee \neg B) \leftrightarrow \neg(A \wedge B)$$

$$\neg(\neg A \vee \neg B) \leftrightarrow (A \wedge B)$$

$$\neg(A \vee \neg B) \leftrightarrow (\neg A \wedge B)$$

$$\neg(\neg A \vee B) \leftrightarrow (A \wedge \neg B)$$

Negation

$$A \vee \neg A$$

$$\neg A \vee A$$

$$(A \rightarrow \neg A) \leftrightarrow \neg A$$

$$(\neg A \rightarrow A) \leftrightarrow A$$

Implication

$$\begin{aligned}(\neg A \vee B) &\leftrightarrow (A \rightarrow B) \\(A \wedge \neg B) &\leftrightarrow \neg(A \rightarrow B) \\(\neg B \rightarrow \neg A) &\leftrightarrow (A \rightarrow B)\end{aligned}$$

Simplification

$$\begin{aligned}A \wedge B &\rightarrow A \\A \wedge B &\rightarrow B\end{aligned}$$

$$\begin{aligned}\forall x(A) &\rightarrow (A)[x = 1] \\ \forall x(A) &\rightarrow (A)[x = 2] \\ \forall x(A) &\rightarrow (A)[x = 3] \\ \forall x(A) &\rightarrow (A)[x = 4] \\ \forall x(A) &\rightarrow (A)[x = 5]\end{aligned}$$

$$\begin{aligned}(A \leftrightarrow B) &\rightarrow (A \rightarrow B) \\(A \leftrightarrow B) &\rightarrow (B \rightarrow A)\end{aligned}$$

$$\begin{aligned}\exists x(A \wedge B) &\rightarrow \exists x(A) \\ \exists x(A \wedge B) &\rightarrow \exists x(B)\end{aligned}$$

$$\begin{aligned}\forall x(A \wedge B) &\rightarrow \forall x(A) \\ \forall x(A \wedge B) &\rightarrow \forall x(B)\end{aligned}$$

Addition

$$\begin{aligned}A &\rightarrow (A \vee B) \\ B &\rightarrow (A \vee B)\end{aligned}$$

$$\begin{aligned}(A)[x = 1] &\rightarrow \exists x(A) \\ (A)[x = 2] &\rightarrow \exists x(A) \\ (A)[x = 3] &\rightarrow \exists x(A) \\ (A)[x = 4] &\rightarrow \exists x(A) \\ (A)[x = 5] &\rightarrow \exists x(A)\end{aligned}$$

$$\exists x(A \wedge B) \rightarrow \exists x(A) \wedge \exists x(B)$$

$$\begin{aligned}\forall x(A) &\rightarrow \forall x(A \vee B) \\ \forall x(B) &\rightarrow \forall x(A \vee B)\end{aligned}$$

$$\begin{aligned}\forall x(\neg A) &\rightarrow \forall x(A \rightarrow B) \\ \forall x(B) &\rightarrow \forall x(A \rightarrow B)\end{aligned}$$

Excluded Middle

$$\begin{aligned}A \vee \neg A \vee B \\ \neg A \vee A \vee B \\ B \vee A \vee \neg A\end{aligned}$$

$$\begin{aligned}
& B \vee \neg A \vee A \\
& A \vee B \vee \neg A \\
& \neg A \vee B \vee A
\end{aligned}$$

$$\begin{aligned}
& A \vee (\neg A \vee B) \\
& \neg A \vee (A \vee B) \\
& B \vee (A \vee \neg A) \\
& B \vee (\neg A \vee A) \\
& A \vee (B \vee \neg A) \\
& \neg A \vee (B \vee A)
\end{aligned}$$

Quantifier Expressions

$$\begin{aligned}
& \forall x(\top) \\
& \forall x\{\}(A) \\
& \forall x(A) \wedge \forall x(B) \leftrightarrow \forall x(A \wedge B) \\
& \exists x(A) \vee \exists x(B) \leftrightarrow \exists x(A \vee B) \\
& \exists x\exists y(A) \leftrightarrow \exists y\exists x(A) \\
& \forall x\forall y(A) \leftrightarrow \forall y\forall x(A)
\end{aligned}$$

D.2 Contradictions

Truth-table entries

$$\begin{aligned}
& \perp \\
& \neg\top \\
& \top \wedge \perp \\
& \perp \wedge \top \\
& \perp \wedge \perp \\
& \perp \vee \perp \\
& \top \rightarrow \perp \\
& \top \leftrightarrow \perp \\
& \perp \leftrightarrow \top
\end{aligned}$$

Identity

$$\begin{aligned}
& A \wedge \perp \\
& \perp \wedge A
\end{aligned}$$

Double Negation

$$\neg\neg\perp$$

Negation

$$\begin{aligned}
& A \wedge \neg A \\
& \neg A \wedge A \\
& \neg A \leftrightarrow A
\end{aligned}$$

Quantifier Expressions

$\exists x(\perp)$
 $\exists x\{\}(A)$

Predicates

Blue(1) & Red(1)
 Blue(1) & Yellow(1)
 Red(1) & Blue(1)
 Red(1) & Yellow(1)
 Yellow(1) & Blue(1)
 Yellow(1) & Red(1)

Blue(2) & Red(2)
 Blue(2) & Yellow(2)
 Red(2) & Blue(2)
 Red(2) & Yellow(2)
 Yellow(2) & Blue(2)
 Yellow(2) & Red(2)

Blue(3) & Red(3)
 Blue(3) & Yellow(3)
 Red(3) & Blue(3)
 Red(3) & Yellow(3)
 Yellow(3) & Blue(3)
 Yellow(3) & Red(3)

Blue(4) & Red(4)
 Blue(4) & Yellow(4)
 Red(4) & Blue(4)
 Red(4) & Yellow(4)
 Yellow(4) & Blue(4)
 Yellow(4) & Red(4)

Blue(5) & Red(5)
 Blue(5) & Yellow(5)
 Red(5) & Blue(5)
 Red(5) & Yellow(5)
 Yellow(5) & Blue(5)
 Yellow(5) & Red(5)

D.3 Non-contradictions

$A \rightarrow B$
 $A \vee B$
 $B \vee A$