J Log Lang Inf (2013) 22:115–137
DOI 10.1007/s10849-012-9168-y

Reasoning About Truth in First-Order Logic

Claes Strannegård · Fredrik Engström · Abdul Rahim Nizamani · Lance Rips

Published online: 13 February 2013

Abstract First, we describe a psychological experiment in which the participants were asked to determine whether sentences of first-order logic were true or false in finite graphs. Second, we define two proof systems for reasoning about truth and falsity in first-order logic. These proof systems feature explicit models of cognitive resources such as declarative memory, procedural memory, working memory, and sensory memory. Third, we describe a computer program that is used to find the smallest proofs in the aforementioned proof systems when capacity limits are put on the cognitive resources. Finally, we investigate the correlation between a number of mathematical complexity measures defined on graphs and sentences and some psychological complexity measures that were recorded in the experiment.

Keywords First-order logic · Proof system · Bounded cognitive resources · Truth

C. Strannegård (✉) · F. Engström
Department of Philosophy, Linguistics and Theory of Science,
University of Gothenburg, Gothenburg, Sweden
e-mail: claes.strannegard@gu.se

C. Strannegård
Department of Applied Information Technology, Chalmers University of Technology,
Gothenburg, Sweden

A. R. Nizamani
Department of Applied Information Technology, University of Gothenburg,
Gothenburg, Sweden

L. Rips
Department of Psychology, Northwestern University, Evanston, IL, USA

1 Introduction

What truths are humans able to identify in the case of first-order logic (FO) on finite models? This is a fundamental question of logic, linguistics, and psychology, which nevertheless remains largely unexplored. In this paper, we study this question systematically using psychological experiments and computational models. We combine elements of proof theory and cognitive psychology and extend earlier work on propositional logic (Strannegård et al. 2010).

1.1 Psychological Complexity

Human reasoning concerning FO truth lends itself perfectly to exploration using standard methods of experimental psychology. This holds also for other model-theoretic logics (Ebbinghaus 1985). Fix a vocabulary τ and let MOD be the set of finite τ-structures for which the domains are subsets of some fixed denumerable set. Also, let SEN be the set of FO τ-sentences and define

$$\mathsf{Truth} = \{(M, A) \in \mathrm{MOD} \times \mathrm{SEN} | M \models A\}.$$

A psychological experiment pertaining to Truth and a given participant can be set up by specifying the experiment's procedure and a finite set of test items TEST $\subset$ MOD $\times$ SEN. An example of such a test item is given in Fig. 1. An experiment of this type yields response times and correctness data for the items of TEST. It also yields a set POS $\subseteq$ TEST consisting of those elements of TEST that the participant classified as members of Truth. The sets POS and Truth generate a division of TEST into four subsets: true positives, false positives, true negatives, and false negatives. These experimental data can then be approximated using computational models, for which the predictive power can be evaluated using statistical measures such as the (Pearson) correlation.

1.2 Mathematical Complexity

What factors influence the difficulty for a human to determine whether $M \models A$ for a finite model M and FO sentence A? Are they the properties of the sentence, the model, or some combination of such properties?

The properties of the sentences are clearly important. For instance, the truth-value of $\exists x E(x, x)$ is generally easier to determine than the truth-value of $\forall x \exists y E(x, y)$. Which properties of sentences are important here? Is it length, quantifier depth, parse tree depth, number of negations, type of connectives, a combination of these properties, or something else?

It is also clear that the properties of the models matter. For instance, the difficulty of determining the truth-value of $\forall x \exists y E(x, y)$ clearly depends on the properties of the model. Which properties of models are important in this connection? Is it cardinality, number of edges, some version of Kolmogorov complexity, a combination of these, or something else?

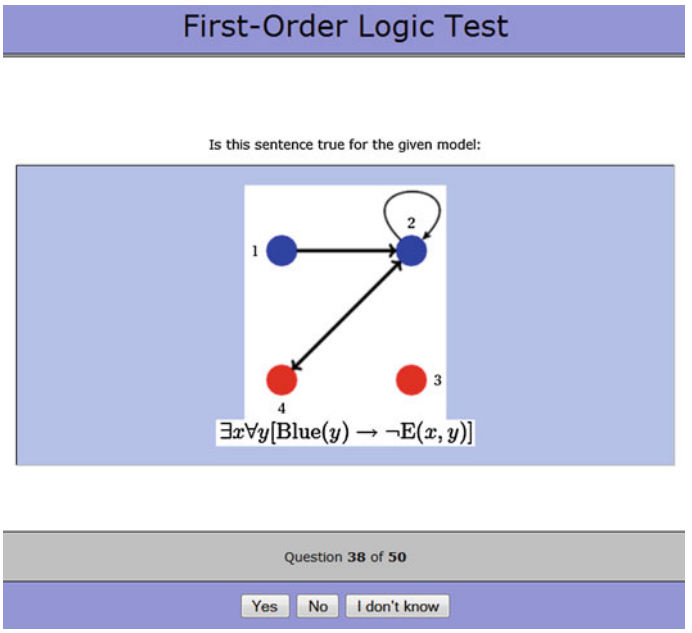


Fig. 1 Screenshot of the interface used for presenting the test items. The correct answer to the item shown is "Yes"

One strategy for predicting the difficulty of determining truth is to combine n complexity measures on sentences with m complexity measures on models using some function f (e.g., a polynomial) of $n + m$ variables. Despite the apparent generality of this strategy, it might be inadequate regardless of the choice of f. In fact, if the interplay between sentences and models is more intricate than that, then this strategy is bound to fail.

An entirely different strategy is to assume that truth-values are computed and that the properties of such computations are the most promising indicators of difficulty. This is the strategy we use in this paper.

1.3 Models of Human Reasoning

Cognitive architectures, such as SOAR (Laird et al. 1987), ACT-R (Anderson and Lebiere 1998), Clarion (Sun 2007), and Polyscheme (Cassimatis 2002), are computational models that have been used to model human reasoning in a wide range of domains. They typically include explicit models of cognitive resources, such as working memory, sensory memory, declarative memory, and procedural memory; these cognitive resources are known to be bounded in various ways, e.g., with respect to capacity, duration, and access time (Kosslyn and Smith 2006).

1.4 Models of Logical Reasoning

In cognitive psychology, several models of logical reasoning have been presented in the mental logic tradition (Braine and O'Brien 1998) and the mental models tradition

(Johnson-Laird 1983). Computational models in the mental logic tradition are commonly based on natural deduction systems (Prawitz 1965); the PSYCOP system (Rips 1996) is one example. The analytical focus of the mental models tradition has been on exploring particular examples rather than developing general computational models.

Human reasoning in the domain of logic is considered from several perspectives in Adler and Rips (2008) and Holyoak and Morrison (2005). Stenning and van Lambalgen (2008) consider logical reasoning both in the laboratory and "in the wild" and investigate how problems formulated in natural language and situated in the real world are interpreted (reasoning to an interpretation) and then solved (reasoning from an interpretation).

Formalisms of logical reasoning include natural deduction (Prawitz 1965; Jaskowski 1934; Gentzen 1969), sequent calculus (Negri and von Plato 2001), natural deduction in sequent calculus style (Negri and von Plato 2001), natural deduction in linear style (Fitch 1952; Geuvers and Nederpelt 2004), and analytic tableaux (Smullyan 1995). Several formalisms have also emerged in the context of automated reasoning, e.g., Robinson's resolution system (2001) and Stålmarck's system (2000). None of these formalisms represent working memory explicitly, and most of them were constructed for purposes other than modeling human reasoning.

We suspect that working memory could also be a critical cognitive resource in the special case of logical reasoning (Gilhooly et al. 1993; Hitch and Baddeley 1976; Toms et al. 1993). Therefore, we will define our own proof system that includes an explicit model of working memory.

1.5 Structure of the Paper

In Sect. 2, we report the results of a psychological experiment concerning FO truth. In Sects. 3 and 4, we present proof systems for showing FO truth and FO falsity, respectively. In Sect. 5, we present resource-bounded versions of these proof systems. In Sect. 6, we present a number of mathematical complexity measures, some of which are defined in terms of the resource-bounded proof systems. In Sect. 7, we compare the psychological complexity measures that were obtained in the experiment with the mathematical complexity measures. Section 8 contains a discussion and Sect. 9, finally, presents some conclusions.

2 Experiment

In this section, we describe a psychological experiment concerning Truth.

2.1 Participants

The participants in our experiment were ten computer science students from Gothenburg, Sweden, whom were invited through email. These students had previously studied FO in their university education. They belonged to various nationalities, were aged 20–30 years, and included one woman and nine men.

2.2 Material

Fifty test items were prepared for the experiment. Each item consisted of a finite graph and an FO sentence. The task was to determine whether the sentence was true in the graph. A screenshot of the graphical user interface is shown in Fig. 1. The test comprised 24 true items and 26 false ones, which were presented in an order that was randomized for each participant.

The logical symbols used in the experiment were the quantifiers $\forall$ and $\exists$ and the connectives $\neg$, $\wedge$, $\vee$, $\rightarrow$, and $\leftrightarrow$. The vocabulary τ included the binary predicate $E(x, y)$ (for edges); the unary predicates $\mathrm{Red}(x)$, $\mathrm{Blue}(x)$, and $\mathrm{Yellow}(x)$ (for colors); and the constants 1, 2, 3, and 4 (for nodes). Let $L(\tau)$ be the set of FO-formulas defined in the usual way over the vocabulary τ.

A total of 50 test items, consisting of pairs of graphs and $L(\tau)$-sentences, were prepared. They were selected manually on the basis of estimated level of difficulty from randomly generated lists of graphs and sentences. All nodes were labeled with unique numbers and colored either yellow, red, or blue.

Some of the test items are given in Appendix A and the full list can be found in Nizamani (2010).

2.3 Procedure

The experiment was conducted in a computer laboratory at the Department of Applied Information Technology, University of Gothenburg. Each participant was assigned an individual computer terminal. A short practice session was conducted before the test to familiarize the participants with some sample problems. The experiment was conducted in a 25-min session, followed by a 10-min break, followed by another 25-min session.

The answers and the response times were recorded for each participant and each item. Two aggregated complexity measures were computed for each item. The *accuracy* is the proportion of participants who answered the item correctly. The *latency* is the median response time among the participants who answered the item correctly. The median was used, rather than the mean, to reduce the effects of extreme response times due to external disturbances.

3 The Proof System $\mathcal{T}_M$

In this section, we describe a proof system $\mathcal{T}_M$ for proving sentences to be true in a fixed finite model M. Please observe that this system is not a classical proof system in which only logical truths can be derived, but a system in which all truths in some fixed model M are derivable.

The system $\mathcal{T}_M$ is a rather straightforward rewrite system with some modifications and extensions. The proofs are *linear* sequences of sentences. The system is *local* in the sense that checking whether a sequence of sentences is a proof can be done by only looking at two consecutive sentences at a time. The proofs are also *goal-driven*,

which means that they start with a proof goal, i.e., the desired conclusion, and then use rules to reduce the goal to the true statement $\top$.

The proof system has two ingredients, axioms and rules. The axioms model declarative memory content and visual information about models and sentences, whereas the rules model the procedural memory. The main rule (Substitution) is based on the principle of compositionality, which is that meaning (and truth-value, in particular) is preserved under the substitution of logically equivalent components. The axioms are used as side-conditions to this rule. For example, substitution of A for B is only allowed if $A \leftrightarrow B$ is an axiom. Substitution is a *deep* rule, which means that it allows subformulas appearing deep down in the parse-tree to be substituted.

3.1 Formulas

We shall now enrich our set of formulas $L(\tau)$ by adding

(i) bounded quantifiers $\forall^{\Omega}$ and $\exists^{\Omega}$, where Ω is a set of constant symbols in τ and $\forall^{\Omega} x\, A$ has the intended meaning $\forall x(\bigvee_{c\in\Omega} x = c \rightarrow A)$;
(ii) abstraction boxes for modeling visual perception of formulas, $[\![A]\!]$; and
(iii) contexts for assigning values to variables, $[x_1 = c_1, \ldots, x_k = c_k]$.

Definition 1 (Formula) The set of $L^*(\tau)$-formulas is defined by the following BNF grammar:

$$\begin{aligned} A &::= \text{Atom} \mid \neg A \mid A \cdot A \mid QxA \mid A[x_1 = c_1, \ldots, x_k = c_k] \\ \text{Atom} &::= P(t_1, \ldots, t_k) \mid [\![B]\!] \end{aligned}$$

where $\cdot$ is one of $\neg$, $\wedge$, $\vee$, $\rightarrow$, or $\leftrightarrow$; Q is either $\forall$, $\exists$, $\forall^{\Omega}$, or $\exists^{\Omega}$, Ω is a set of constant symbols in τ; $c_1, \ldots, c_k$ are constant symbols in τ; P is a predicate in τ; and B is an $L(\tau)$-formula.

Definition 2 (Satisfaction relation) The *satisfaction relation* for formulas in $L^*(\tau)$ is defined in the standard Tarskian way with the following extra clauses, where s is an assignment:

- $M \models_s \exists^{\Omega} x\, A$ iff $M \models_s \exists x(\bigvee_{c\in\Omega} x = c \wedge A)$,
- $M \models_s \forall^{\Omega} x\, A$ iff $M \models_s \forall x(\bigvee_{c\in\Omega} x = c \rightarrow A)$,
- $M \models_s [\![A]\!]$ iff $M \models_s A$, and
- $M \models_s A[x_1 = c_1, \ldots, x_k = c_k]$ iff $M \models_{s[c_1/x_1,\ldots,c_k/x_k]} A$.

Next, we define what it means for a relation symbol R in a formula to be substituted by a formula.

Definition 3 (Substitution) If R is a relation symbol of arity k, A a formula in $L^*(\tau \cup \{R\})$ and B a formula in $L^*(\tau)$ with the free variables $\bar{x} = x_1, \ldots, x_k$, we define the substitution $A[B(\bar{x})/R]$ to be the formula we obtain from A by substituting the leftmost occurrence of the symbol R with $B[t_1, \ldots, t_k/x_1, \ldots, x_k]$, where R occurs as $R(t_1, \ldots, t_k)$ and t_i is free for x_i in B (if not, the notation $A[B(\bar{x})/R]$ is undefined).

Observe that substitutions are not carried out inside abstraction boxes $[\![C]\!]$.

3.2 Axioms

The axiom set Γ is constructed as the union of three sets: the logical axioms, the sentence axioms and the model axioms.

3.2.1 Logical Axioms

The set of logical axioms is denoted Γ_T. The members of this set are the logical truths listed in Appendix B. This list was compiled from a number of textbooks on basic logic, including (Huth and Ryan 2004).

3.2.2 Sentence Axioms

The second ingredient of Γ comes from the idea to include parsing of a formula in the proof system itself. When we must check the truth of a formula, we first must parse the formula. This action is modeled in the system using abstraction boxes, which are considered black boxes that the agent cannot look inside. By using the rules of the proof system, the agent may unwind the formula one step at a time. Formally, we add one *abstraction box* for each formula in the language. The abstraction box corresponding to the formula A is denoted by $[\![A]\!]$. These boxes are in many respects similar to atomic formulas; the free variables of the abstraction box $[\![A]\!]$ are defined to be the same as the free variables of A. This idea is closely related to the template logic introduced in Engström (2002), which was constructed to understand non-standard formulas in non-standard models of arithmetic.

In a bounded version of the proof system that will be introduced later, we will restrict the complexity of the formulas that may be used in a proof. Thus, the abstraction boxes may be used to reduce this complexity. The drawback of this reduction in complexity is that the substitution rule (see Sect. 3.3.1) may not substitute inside abstraction boxes.

To include parsing in the system, we add, for each pair of formulas A and B of $L(\tau)$ and each variable x, the following axioms to the set Γ_S of sentence axioms:

- If A is atomic, $[\![A]\!] \leftrightarrow A$.
- $[\![\neg A]\!] \leftrightarrow \neg [\![A]\!]$
- $[\![A \cdot B]\!] \leftrightarrow [\![A]\!] \cdot [\![B]\!]$, where $\cdot$ is one of $\vee$, $\wedge$, $\rightarrow$, and $\leftrightarrow$.
- If A is atomic, $[\![A \cdot B]\!] \leftrightarrow A \cdot [\![B]\!]$, where $\cdot$ is one of $\vee$, $\wedge$, $\rightarrow$, and $\leftrightarrow$ (similarly for B).
- $[\![Qx A]\!] \leftrightarrow Qx [\![A]\!]$, where Q is one of $\forall$, $\forall^{\Omega}$, $\exists$, and $\exists^{\Omega}$.
- If A is atomic, $[\![Qx A]\!] \leftrightarrow Qx A$, where Q is one of $\forall$, $\forall^{\Omega}$, $\exists$, and $\exists^{\Omega}$.

Thus, by using these axioms and the substitution rule, we can replace a template symbol in a formula by a more complex formula. By repeating this procedure, we may eliminate all abstraction boxes from a formula, thus replacing $[\![A]\!]$ by A. However, this result is achieved only with a considerable increase in complexity.

3.2.3 Model Axioms

Let M be a finite structure in which all elements are named by some finite set of constants Ω_0. To analyze the quantifiers in formulas, we require axioms that specify the range of quantifiers and the truth of quantifier-free sentences. If A is an $L^*(\tau)$-formula (including abstraction boxes), we denote the corresponding formula without abstraction boxes by A'. The set Γ_M consists of the following formulas:

1. A and $A \leftrightarrow \top$, whenever A' is a quantifier-free formula that is true in M.
2. $\neg A$ and $A \leftrightarrow \bot$, whenever A' is a quantifier-free formula that is false in M.
3. $\forall x\, A \leftrightarrow \forall^{\Omega_0} x\, A$.
4. $\exists x\, A \leftrightarrow \exists^{\Omega_0} x\, A$.
5. $\forall^{\Omega} x\, A \leftrightarrow \forall^{\Omega\setminus\{c\}} x\, A$, where $M \models A'[x = c]$ and A' is quantifier-free.
6. $\exists^{\Omega} x\, A \leftrightarrow \exists^{\Omega\setminus\{c\}} x\, A$, where $M \not\models A'[x = c]$ and A' is quantifier-free.
7. $\forall^{\Omega} x\, (B \rightarrow A) \leftrightarrow \forall^{\Omega'} x\, A$, where $\Omega' = \{a | M \models B[x = a]\}$ and B is an atomic formula.
8. $\exists^{\Omega} x\, (B \wedge A) \leftrightarrow \exists^{\Omega'} x\, A$, where $\Omega' = \{a | M \models B[x = a]\}$ and B is an atomic formula.

In the above formulas, Ω is any set of constants.

The set of axioms Γ is defined as the union of Γ_T, Γ_S, and Γ_M (which depends on the model M).

3.3 Rules

In this subsection, we present the rules of $\mathcal{T}_M$.

3.3.1 Substitution

This is a deep rule in the sense that formulas can be substituted deep in the parse tree of a sentence. Only one occurrence of a formula can be substituted at a time:

$$\frac{A[B(\bar{x})/R]}{A[C(\bar{x})/R]}\ \text{LE/FI/MI}$$

This rule may only be applied if $C \leftrightarrow B \in \Gamma$ and $\mathrm{FV}(C) = \mathrm{FV}(B)$. We label the rule by LE (Logical Equivalence), FI (Formula Inspect), or MI (Model Inspect) when the equivalence $B \leftrightarrow C$ comes from Γ_T, Γ_S, or Γ_M, respectively.

In the case of MI, we allow all contexts in A to be used for determining whether $C \leftrightarrow B$ is in Γ_M. I.e., if γ is the list of all contexts in A then B may be replaced by C if $(C\gamma) \leftrightarrow (B\gamma) \in \Gamma_M$.[1] For example in a model where $P(c)$ holds we may use MI to deduce $(\top \wedge B)[x = c]$ from $(P(x) \wedge B)[x = c]$.

In short, Substitution enables us to deduce $A[C(\bar{x})/R]$ from $A[B(\bar{x})/R]$ whenever $C \leftrightarrow B \in \Gamma$ and C and B have the same free variables.

[1] This can be defined in a precise manner, but we omit the technical details here.

3.3.2 Strengthening

We may replace the goal to show B to be true with the goal to show A to be true whenever we know $A \to B$ to be a logical truth:

$$\frac{B}{A}\ \mathrm{S}$$

To apply this rule, we require that $A \to B \in \Gamma_T$.

3.3.3 Truth Recall

If we have derived a formula that we know to be a logical truth, we have succeeded:

$$\frac{A}{\top}\ \mathrm{TR}$$

To apply this rule, we require that $A \in \Gamma_T$.

This concludes the definition of the proof system $\mathcal{T}_M$.

3.4 Proofs

Definition 4 (Proof in $\mathcal{T}_M$) Suppose $M \in \mathrm{MOD}$ and $A \in \mathrm{SEN}$. A *proof* of A in $\mathcal{T}_M$ is a sequence of sentences $(A_0, A_1, \ldots, A_n)$ such that $A_0 = [\![A]\!]$, $A_n = \top$, and A_{i+1} follows from A_i by one of the rules of $\mathcal{T}_M$.

By letting the first sentence in the sequence be $[\![A]\!]$ rather than A, we aim to model the idea that an individual who is presented with a sentence perceives the structure of the sentence gradually, starting from perceiving nothing and possibly ending before all the details have been perceived. Formula inspection can be used for gradually perceiving and parsing the sentence.

As an example of a proof in $\mathcal{T}_M$, let us prove that the sentence given in Fig. 1 is true in the model given in the same figure:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[\![\exists^{\{1,2,3,4\}}x \forall^{\{1,2,3,4\}}y(Blue(y) \to \neg E(x,y))]\!]}{\exists^{\{1,2,3,4\}}x[\![\forall^{\{1,2,3,4\}}y(Blue(y) \to \neg E(x,y))]\!]}\ \mathrm{FI}}{[\![\forall^{\{1,2,3,4\}}y(Blue(y) \to \neg E(x,y))]\!][x=3]}\ \mathrm{S}\ (A[x=3] \to \exists x A)}{(\forall^{\{1,2,3,4\}}y[\![Blue(y) \to \neg E(x,y)]\!])[x=3]}\ \mathrm{FI}}{(\forall^{\{2,3,4\}}y[\![Blue(y) \to \neg E(x,y)]\!])[x=3]}\ \mathrm{MI}}{(\forall^{\{3,4\}}y[\![Blue(y) \to \neg E(x,y)]\!])[x=3]}\ \mathrm{MI}}{(\forall^{\{4\}}y[\![Blue(y) \to \neg E(x,y)]\!])[x=3]}\ \mathrm{MI}}{(\forall^{\emptyset}y[\![Blue(y) \to \neg E(x,y)]\!])[x=3]}\ \mathrm{MI}}{\top}\ \mathrm{TR}\ (\forall^{\emptyset}A)$$

Note that the abstraction boxes may be removed in the beginning of the proofs by zooming in on the formula using iterated applications of Substitution (FI). Abstraction boxes are strictly speaking unnecessary at this point, but they will play an important role in Sect. 5, in which bounded versions of proof systems will be considered.

3.5 Properties

The system $\mathcal{T}_M$ includes simple models of the following cognitive resources:

- declarative memory, which is modeled by the logical axioms in the set Γ_T;
- procedural memory, which is modeled by the rules;
- sensory memory, which is modeled as a buffer to hold sentence axioms from the set Γ_S (modeling visual perception of sentence structure);
- working memory, which is modeled as a buffer to hold temporary proof goals.

Now, let us show the adequacy of $\mathcal{T}_M$ for Truth.

Proposition 1 *Suppose $M \in MOD$ and $A \in SEN$. Then, $M \models A$ iff A is provable in $\mathcal{T}_M$.*

Proof Right-to-left is soundness: We prove that in any proof of $\mathcal{T}_M$, if A occurs, then $M \models A$. This is done by induction starting from the bottom; the trivial base case is when A is $\top$. For the induction step, all we must check is that truth is preserved (in the sense that if A may be deduced from B and $M \models A$, then $M \models B$) by the rules in $\mathcal{T}_M$. This is straightforward.

Left-to-right is completeness: Suppose $M \models A$. Note that by essentially using Substitution (FI) followed by Substitution (MI), we may reduce any sentence A to a propositional formula in which only $\top$ and $\bot$ are atomic formulas. Regardless of the details of this propositional formula, we can then use Substitution (LE) to produce strictly shorter formulas at each step until either $\top$ or $\bot$ is reached. Because $M \models A$, however, the soundness of the system forces the proof to end with $\top$.

4 The Proof System $\mathcal{F}_M$

In this section, we describe a proof system $\mathcal{F}_M$ for showing sentences to be false in a fixed finite model M. The formulas and axioms of $\mathcal{F}_M$ are the same as for $\mathcal{T}_M$.

4.1 Rules

Now, let us define the rules of $\mathcal{F}_M$.

4.1.1 Substitution

This rule is identical to the Substitution rule of $\mathcal{T}_M$.

4.1.2 Weakening

We may replace the goal of showing B to be false with the goal of showing A to be false whenever we know $B \rightarrow A$ to be a logical truth:

$$\frac{B}{A}\ \mathrm{W}$$

To apply this rule, we require that $B \rightarrow A \in \Gamma_T$.

4.1.3 Falsity Recall

This rule makes use of a set Γ_F, which contains contradictions only. We omit the exact definition of Γ_F, which is a list of textbook contradictions in the style of Appendix B.

If we have derived a formula which we know to be a contradiction, we have succeeded:

$$\frac{A}{\bot}\ \text{FR}$$

To apply this rule, we require that $A \in \Gamma_F$.

4.2 Properties

Definition 5 (Proof in $\mathcal{F}_M$) Suppose that $M \in$ MOD and $A \in$ SEN. A *proof* of A in $\mathcal{F}_M$ is a sequence of sentences $(A_0, A_1, \ldots, A_n)$ such that $A_0 = [\![A]\!]$, $A_n = \bot$, and A_{i+1} follows from A_i by one of the rules of $\mathcal{F}_M$.

Now, let us prove the adequacy of $\mathcal{F}_M$ for showing falsity in M.

Proposition 2 *Suppose $M \in$ MOD and $A \in$ SEN. Then, $M \not\models A$ iff A is provable in $\mathcal{F}_M$.*

Proof This proof is analogous to the proof of Proposition 1.

5 The Bounded Proof Systems $\mathcal{BT}_M$ and $\mathcal{BF}_M$

In this section, we define resource-bounded versions of the proof systems $\mathcal{T}_M$ and $\mathcal{F}_M$. For this purpose, we require a precise definition of sentence length.

Definition 6 (Formula length) The *length* $|A|$ of an $L^*(\tau)$-formula A is defined as follows:

- $|A| = 1$ when A is atomic.
- $|[\![A]\!]| = 1$.
- $|A[x_1 = c_1, \ldots, x_k = c_k]| = 1 + |A|$.
- $|\neg A| = 1 + |A|$.
- $|A \cdot B| = 1 + |A| + |B|$, where $\cdot$ is either $\vee$, $\wedge$, $\rightarrow$ or $\leftrightarrow$.
- $|Qx A| = 2 + |A|$, where Q is either $\forall$, $\forall^{\Omega}$, $\exists$, or $\exists^{\Omega}$.

Now we can define our two bounded proof systems.

Definition 7 ($\mathcal{BT}_M$ and $\mathcal{BF}_M$) Let $\mathcal{BT}_M$ be the proof system obtained from $\mathcal{T}_M$ by adding the following restrictions on the proofs:

- Working memory limit. The maximum length of a sentence that can appear in a proof is 8.
- Sensory memory limit. The rule Substitution (FI) is only allowed on sentence axioms of Γ_S of maximum length 7.

The proof system $\mathcal{BF}_M$ is defined analogously from $\mathcal{F}_M$.

In the bounded proof systems, the abstraction boxes play an important role because they enable certain sentences to be provable that would not be provable without them. For instance, note that the FO sentence $\top \vee A$, where $|A| > 6$ is valid and that this validity can be deduced without looking inside the right disjunct. This sentence is not allowed to appear in a proof in $\mathcal{BT}_M$, however, because its length exceeds 8. However, we have the following proof in $\mathcal{BT}_M$:

$$\dfrac{\dfrac{[\![\top \vee A]\!]}{\top \vee [\![A]\!]}\ \text{FI}}{\top}\ \text{TR}\ (\top \vee A)$$

6 Mathematical Complexity Measures

In this section we define a number of mathematical complexity measures on items (M, A). The complexity measures are as follows (the ranges refer to the items appearing in the experiment).

1. Sentence length. The length of A. Range: 5–9.
2. Quantifier count. The number of quantifiers appearing in A. Range: 1–3.
3. Negation count. The number of negations appearing in A. Range: 0–2.
4. Cardinality. The number of nodes of M. Range: 3–4.
5. Edge count. The number of edges of M. Range: 3–6.
6. Linear combination. A linear combination of sentence length, cardinality, and sentence length * cardinality, whose coefficients can be fitted to experimental data.
7. Working memory. The minimum size of the WM required for proving A in $\mathcal{BT}_M$ or $\mathcal{BF}_M$. Range: 3–8.
8. Proof length. The minimum number of steps of a proof of A in $\mathcal{BT}_M$ or $\mathcal{BF}_M$. Range: 4–38.
9. Proof size. The minimum size of a proof of A in $\mathcal{BT}_M$ or $\mathcal{BF}_M$, i.e. the minimum sum of the lengths of the formulas appearing in the proof. Range: 6–164.

Let us provide some motivation why these particular complexity measures were chosen.

Measures 1–5 are standard complexity measures in logic. Measure 6 (Linear combination) illustrates the possibility of combining such complexity measures in various ways, e.g. via polynomials. As suggested in Sect. 1.2, there may be a priori arguments for believing that these complexity measures are inadequate for the present experiment. Measure 7 (Working memory) models the maximum strain on the working memory and measure 8 (Proof length) models the length of the shortest train of thought leading to the desired conclusion.

Measure 9 (Proof size) is based on the idea that latency in the context of logical reasoning depends on some notion of computational workload. By identifying the working memory load with formula length, we get proof size as a measure of computational workload. Thus measure 9 models the minimum amount of data that must flow through the working memory for the problem to be solved.

Table 1 Mean latency and mean accuracy for true and false items

	True	False
Latency (seconds)	33	37
Accuracy (%)	74	67

Table 2 Correlations between latency and some mathematical complexity measures

	True	False
Sentence length	0.35	0.63
Quantifier count	0.42	0.73
Negation count	0.15	0.12
Cardinality	0.01	−0.02
Edge count	−0.28	0.09
Linear combination	0.38	0.64
Working memory	0.53	0.66
Proof length	0.49	0.76
Proof size	0.54	0.76

Table 3 Correlations between accuracy and some mathematical complexity measures

	True	False
Sentence length	−0.22	−0.02
Quantifier count	−0.22	−0.05
Negation count	−0.33	−0.56
Cardinality	−0.41	0.38
Edge count	−0.04	−0.10
Linear combination	0.46	0.50
Working memory	−0.32	−0.19
Proof length	−0.21	0.13
Proof size	−0.24	0.07

We implemented automated theorem provers for the systems $\mathcal{BT}_M$ and $\mathcal{BF}_M$ in the functional programming language Haskell. Proofs of minimum length and minimum size were generated for all items appearing in the experiment. For instance, the proof appearing in Sect. 3.4 was generated by our theorem prover, as a proof of minimum size.

7 Results

Table 1 shows the mean latency and mean accuracy recorded at the experiment. Table 2 shows correlations between latency and the mathematical complexity measures defined in Sect. 6. Table 3 shows correlations between accuracy and the same complexity measures.

8 Discussion

In this paper, we studied human reasoning in the case of FO truth using standard methods of cognitive psychology. This reflects our view that there is nothing special about problem-solving in the domain of logic, and therefore, it can be explored using ordinary experimental methods and modeled using ordinary models of cognitive psychology that would apply equally well to mental arithmetic, Sudoku, or Minesweeper, for example.

8.1 Comments on the Experiment

In a preliminary investigation, we observed that the difficulty of determining truth was affected by the manner in which the models were presented graphically. For instance, determining whether a graph is complete seems to be simpler if the graph is drawn in a regular fashion. To mitigate this problem, we drew our graphs by placing the nodes equally distributed on a circle.

As is usual in experiments of this type, the answers given by the participants are potentially problematic because guesses, interaction errors (e.g., hitting the wrong button accidentally), and distractions in the experimental situation (e.g., coughing attacks) can affect the individual results substantially. Another potential problem relates to the instructions. In our particular experiment, we learned afterwards that some of the participants thought that if two variables x and y appeared in a sentence, then automatically $x \neq y$. Our instructions failed to address this point.

On the aggregate level, some of the problems on the individual level might partially cancel out, but then new problems arise on the modeling side. In fact, to model experimental data on the aggregated level, one must develop a computational model that represents some sort of average of the participants. Strictly speaking, this might not be possible using our present type of computational model, which was designed for cognitive modeling on the individual level. These factors should be borne in mind when evaluating the results.

8.2 Comments on the Proof Systems

The bounded proof systems described in Sect. 5 can be modified in several ways to suit different human role models. One way is to modify the axioms; a second is to modify the rules; a third is to change the working memory and sensory memory capacities; and a fourth is to add models of other cognitive resources.

Intuitively, proof-size reflects how comprehensive a thought must be, i.e., how much information must be processed to produce the correct answer. Proofs that require more information processing might take longer to process and be more prone to error. One may perhaps think of the smallest proofs as the smartest proofs, relatively to a given repertoire of cognitive resources.

As is often the case with cognitive modeling, this complexity measure can be criticized for being too coarse. For instance, it does not directly reflect the effort of searching for a proof; instead, it reflects the effort required to verify the steps in a proof that has been found. Although the efforts required for finding a proof and verifying

the same proof may be somewhat correlated, this limitation certainly allows for future improvements to the model.

8.3 Comments on the Results

The data in Table 1 indicate that the True items were easier to solve than the False items.

Among the complexity measures analyzed in Table 2, Proof size has the highest correlation with latency, both for True and False items. Second comes the closely related complexity measure Proof length. This might indicate that complexity measures that are defined in terms of computations are more adequate than those defined in terms of standard properties of models and sentences.

Some of the complexity measures on sentences in Table 2 fared relatively well. In Sect. 1 it was argued that in general, properties of models can dramatically affect the difficulty of determining truth. The reason why those complexity measures on sentences were relatively successful in this particular case might be that the models that were used in the experiment were quite homogeneous. In fact, Cardinality varied between 3 and 4, and Edge count varied between 3 and 6. Therefore, the complexity measures that considered only sentences were perhaps not sufficiently challenged in the present experiment.

All of the complexity measures analyzed in Table 3 have relatively low correlations with accuracy. We do not know why the contrast to latency is so pronounced. The correlation values for Negation count stand out here and provide some support to the idea that negations increase the probability of error.

9 Conclusions

In this paper, we presented (i) the results of an experiment pertaining to Truth, (ii) two proof systems for deriving membership and non-membership in Truth using bounded cognitive resources, and (iii) an analysis of the correlation between psychological complexity measures and different mathematical complexity measures, including proof size. The results indicate that proof size was more successful than the other complexity measures in the case of latency. The approach that we use enables predictions about latency to be made for arbitrary elements of Truth. To evaluate the usefulness of this approach, which combines elements of proof theory and cognitive psychology, more experiments are needed, in particular experiments with more heterogeneous test items.

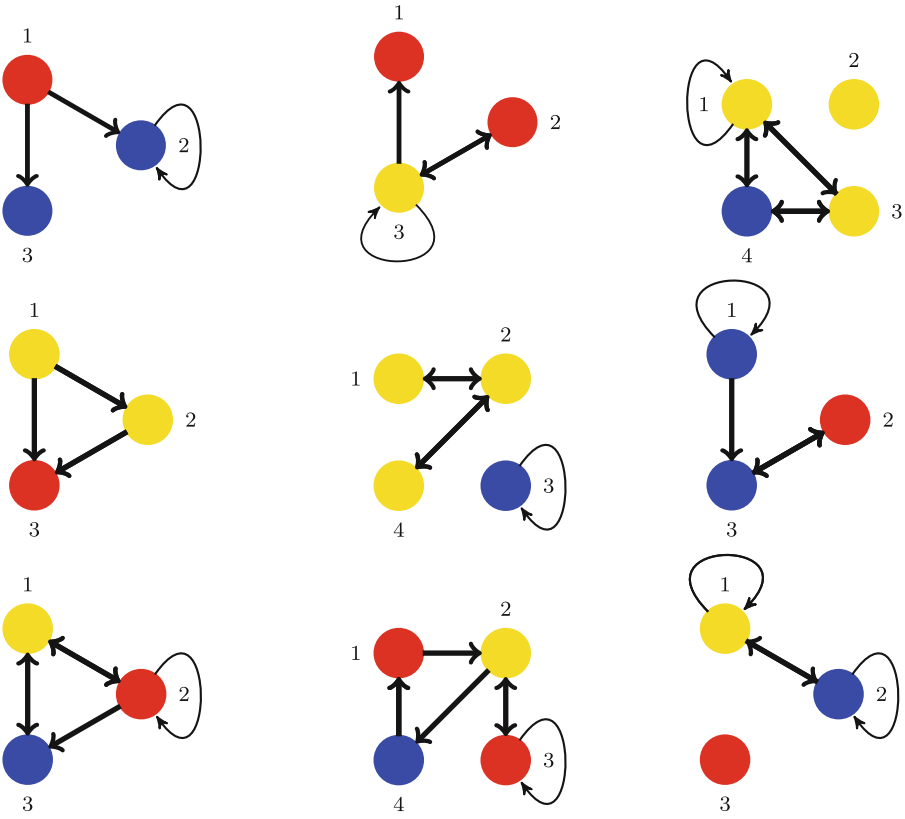
Appendix A: Experimental Data

To enable a deeper understanding of the experiment, let us list items 1–9 of the experiment and give their associated data. A full description of the items used can be found

Table 4 The sentences of items 1–9

Item	Sentence
1	$\exists x\big(Blue(x) \wedge \forall y[Blue(y) \rightarrow E(x,y)]\big)$
2	$\forall x\big(Red(x) \rightarrow \exists y \exists z[E(x,y) \wedge E(y,z) \wedge Red(z)]\big)$
3	$\exists x[Yellow(x) \wedge \forall y \neg E(x,y)]$
4	$\forall x \forall y[\neg E(x,y) \rightarrow Yellow(y)]$
5	$\forall x\big(\exists y[Yellow(y) \wedge E(x,y)] \leftrightarrow Yellow(x)\big)$
6	$\forall x\big([E(x,x) \vee \neg Blue(x)] \rightarrow E(3,x)\big)$
7	$\exists x \exists y[Red(x) \wedge Red(y) \wedge E(x,y) \wedge E(y,x)]$
8	$\exists x \exists y \exists z[E(x,y) \wedge E(y,z) \wedge \neg E(z,x)]$
9	$\forall x[\neg Red(x) \rightarrow \forall y E(x,y)]$

Table 5 The models of items 1–9. On the top row, items 1–3 are shown, and so on



in Nizamani (2010). Tables 4 and 5 list the sentences and models of items 1–9, respectively. Table 6 shows the data recorded at the experiment concerning items 1–9. Table 7, finally, shows the mathematical complexity measures of items 1–9.

Table 6 Response times of the participants P1–P10 on items 1–9

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	M	A
1	47.2	32.4		26.9		49.3	57.6	72.7	10.7	66.6	48.3	8
2	30.3	67.9		35.3	42.6	65.7	54.9		70.5	70.5	60.3	8
3	17.0	23.2	21.2	12.4	47.1		60.0	29.9	30.7	45.4	29.9	9
4				14.0				68.2			41.1	2
5	52.2			31.4			24.5	60.9	71.0	53.3	52.7	6
6	29.1					49.3	73.5	21.5	20.2		29.1	5
7	19.2	29.8		22.9	18.6	19.3	22.2		24.4		22.2	7
8	39.6		25.7	16.4	58.8	37.5	28.2	73.3	26.0		32.8	8
9	21.0	31.5		14.7					48.5		26.3	4

Here M abbreviates median response time and A abbreviates accuracy

Table 7 The mathematical complexity measures of items 1–9

Item	SL	QC	NC	C	EC	WM	PL	PS	Truth-value
1	13	2	0	3	3	5	8	32	False
2	19	3	0	3	3	6	13	71	False
3	11	2	1	4	4	4	9	31	True
4	11	2	1	3	3	4	5	17	False
5	13	2	0	4	4	6	38	164	True
6	13	1	1	3	3	3	4	7	False
7	17	2	0	3	4	4	5	17	True
8	18	3	1	4	5	4	8	27	True
9	11	2	1	3	3	5	8	25	False

Sentence length (SL), Quantifier count (QC), Negation count (NC), Cardinality (C), Edge count (EC), Working memory (WM), Proof length (PL), and Proof size (PS). The last column represents the truth-value of the item

Appendix B: Logical Truths

In this section, the members of the set Γ_T are listed. Note that all formulas listed below are logical truths. Below A, B, and C stand for formulas of $L^*(\tau)$. Remember that if A is an $L^*(\tau)$-formula then A' is the corresponding $L(\tau)$-formula with the abstraction boxes removed. We assume that $\forall x\, A$ includes the case of $\forall^{\Omega} x\, A$ and similarly with $\exists$. For every axiom A appearing in the list, we also include the sentence $A \leftrightarrow \top$ as an element of Γ_T.

B.1 Truth-Table

$\neg\bot$

$\top \wedge \top$

$A \vee \top$

$\top \vee A$

$A \rightarrow \top$

$\bot \rightarrow A$

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

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

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

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

$(\top \rightarrow A) \leftrightarrow A$

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

$(A \leftrightarrow \top) \leftrightarrow A$

$(\top \leftrightarrow A) \leftrightarrow A$

$(\bot \leftrightarrow A) \leftrightarrow \neg A$

$(A \leftrightarrow \bot) \leftrightarrow \neg A$

B.2 Idempotence

$(A \vee A) \leftrightarrow A$

$(A \wedge A) \leftrightarrow A$

$A \rightarrow A$

$A \leftrightarrow A$

B.3 Commutativity

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

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

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

B.4 Associativity

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

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

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

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

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

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

B.5 Distributivity

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

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

B.6 De Morgan

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

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

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

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

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

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

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

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

B.7 Negation

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

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

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

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

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

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

B.8 Excluded Middle

$A \vee \neg A$

$\neg A \vee A$

$A \vee \neg A \vee B$

$\neg A \vee A \vee B$

$B \vee A \vee \neg A$

$B \vee \neg A \vee A$

$A \vee B \vee \neg A$

$\neg A \vee B \vee A$

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

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

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

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

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

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

B.9 Quantifier Expressions

$\forall x \top$

$\forall^{\emptyset} A$

$$\forall^{\Omega} x\, A \leftrightarrow (\forall^{\Omega\setminus\{c\}} A \wedge A[x = c])$$

$$\exists^{\Omega} x\, A \leftrightarrow (\exists^{\Omega\setminus\{c\}} A \vee A[x = c])$$

$$\forall^{\Omega} x\, A \leftrightarrow (\forall^{\Omega\setminus\{c\}} [\![A']\!] \wedge A[x = c])$$

$$\exists^{\Omega} x\, A \leftrightarrow (\exists^{\Omega\setminus\{c\}} [\![A']\!] \vee A[x = c])$$

$$(\forall x\, A \wedge \forall x\, B) \leftrightarrow \forall x (A \wedge B)$$

$$(\exists x\, A \vee \exists x\, B) \leftrightarrow \exists x (A \vee B)$$

$$\exists x \exists y A \leftrightarrow \exists y \exists x A$$

$$\forall x \forall y A \leftrightarrow \forall y \forall x A$$

B.10 Implications

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

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

$$A \rightarrow (A \vee B)$$

$$B \rightarrow (A \vee B)$$

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

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

$$\forall x\, A \rightarrow A[x = c]$$

$$A[x = c] \rightarrow \exists x\, A$$

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

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

$$\forall x (A \wedge B) \rightarrow \forall x\, A$$

$$\forall x (A \wedge B) \rightarrow \forall x\, B$$

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

$$\forall x(\neg A) \rightarrow \forall x(A \rightarrow B)$$

$$\forall x B \rightarrow \forall x(A \rightarrow B)$$

$$\forall x A \rightarrow \forall x(A \vee B)$$

$$\forall x B \rightarrow \forall x(A \vee B).$$

References

Adler, J. E., & Rips, L. J. (2008). *Reasoning: Studies of human inference and its foundations*. Cambridge: Cambridge University Press.

Anderson, J. R., & Lebiere, C. (1998). *The atomic components of thought*. Mahwah, NJ: Lawrence Erlbaum.

Braine, M. D. S., & O'Brien, D. P. (1998). *Mental logic*. UK: L. Erlbaum Associates.

Cassimatis, N. (2002). *Polyscheme: A cognitive architecture for integrating multiple representation and inference schemes*. PhD thesis.

Ebbinghaus, H. D. (1985). *Extended logics: The general framework. Model-theoretic logics* (pp. 25–76).

Engström, F. (2002). *Satisfaction classes in nonstandard models of arithmetic*. Licentiate thesis, Chalmers University of Technology.

Fitch, F. B. (1952). *Symbolic logic: an introduction*. New York: Ronald Press.

Gentzen, G. (1969). Investigation into logical deduction, 1935. In M. E. Szabo (Eds.), *The collected papers of Gerhard Gentzen*. North-Holland Amsterdam.

Geuvers, H., & Nederpelt, R. (2004). Rewriting for Fitch style natural deductions. In *Rewriting techniques and applications*. Springer (pp. 134–154).

Gilhooly, K. J., Logie, R. H., Wetherick, N. E., & Wynn, V. (1993). Working memory and strategies in syllogistic-reasoning tasks. *Memory & Cognition*, *21*(1), 115–124.

Hitch, G. J., & Baddeley, A. D. (1976). Verbal reasoning and working memory. *The Quarterly Journal of Experimental Psychology*, *28*(4), 603–621.

Holyoak, K. J., & Morrison, R. G. (2005). *The Cambridge handbook of thinking and reasoning*. Cambridge: Cambridge University Press.

Huth, M., & Ryan, M. (2004). *Logic in computer science: Modelling and reasoning about systems*. Cambridge, UK: Cambridge University Press.

Jaskowski, S. (1934). The theory of deduction based on the method of suppositions. *Studia Logica*, *1*, 5–32.

Johnson-Laird, P. N. (1983). *Mental models*. Cambridge, MA: Harvard University Press.

Kosslyn, S. M., & Smith, E. E. (2006). *Cognitive psychology: Mind and brain*. Upper Saddle River, NJ: Prentice-Hall Inc.

Laird, J., Newell, A., & Rosenbloom, P. (1987). Soar: An architecture for general intelligence. *Artificial Intelligence*, *33*(3), 1–64.

Negri, S., & von Plato, J. (2001). *Structural proof theory*. Cambridge: Cambridge University Press.

Nizamani, A. R. (2010). *Anthropomorphic proof system for first-order logic*. Masters thesis, Chalmers University of Technology.

Prawitz, D. (1965). Natural deduction. In *A proof-theoretical study, volume 3 of Stockholm studies in philosophy*. Stockholm: Almqvist & Wiksell.

Rips, L. (1996). *The psychology of proof*. Bradford.

Robinson, A., & Voronkov, A. (2001). *Handbook of automated reasoning*. The Netherlands: Elsevier Science.

Sheeran, M., & Stålmarck, G. (January 2000). A tutorial on Stålmarck's proof procedure for propositional logic. *Formal Methods in Systems Design*, *16*(1), 23–58.

Smullyan, R. M. (1995). *Logic, First-Order* (second corrected edition). NewYork: Dover. (Berlin: Heidelberg, New York: First published in 1968 by Springer).

Stenning, K., & van Lambalgen, M. (2008). *Human reasoning and cognitive science*. Cambridge, MA: Bradford Books MIT Press.

Strannegård, C., Ulfsbäcker, S., Hedqvist, D., & Gärling, T. (2010). Reasoning processes in propositional logic. *Journal of Logic, Language and Information*, *19*(3), 283–314.

Sun, R. (2007). The importance of cognitive architectures: An analysis based on CLARION. *Journal of Experimental & Theoretical Artificial Intelligence*, *19*(2), 159–193.

Toms, M., Morris, N., & Ward, D. (1993). Working memory and conditional reasoning. *The Quarterly Journal of Experimental Psychology*, *46*(4), 679–699.