Logic in Action*
An Open Logic Courseware Project

Jan Jaspars and Fernando R. Velázquez-Quesada

Institute for Logic, Language and Computation, University of Amsterdam,
Science Park 904, 1098 XH Amsterdam, The Netherlands
{J.O.M.Jaspars, F.R.VelazquezQuesada}@uva.nl
www.logicinaction.org

Abstract. Today, logic is taught to a much broader academic audience than just the traditional target group of mathematicians and philosophers. Logic in Action provides modern open courseware, in the form of a freely accessible animated e-book, as an extensive teaching tool for this growing student population. The project stresses the methodological merits of logic for a wide range of interdisciplinary sciences.

1 Introduction

In the Netherlands, logic is taught for a much broader group of students than what we were used to in earlier days. There are several reasons for this increase of the academic dissemination of logic. Most importantly, undergraduate programs at Dutch universities have changed considerably during the last two decades. One of the most significant changes has been the introduction of new interdisciplinary studies that offer a wider range of scientific subjects in one program, and therefore tend to attract more and more students who have just left secondary school. They prefer this stepping stone option because it offers them introductory courses on different subjects of study before fine-tuning their academic careers.

Logic, as an interdisciplinary science par excellence, fits very well within the curricula of these kind of studies. At the University of Amsterdam, for example, undergraduate programs like 'exact and social sciences' (Bèta Gamma) and 'liberal arts and sciences', welcoming together more than 300 starting students each year, have incorporated an introductory course in logic within their first year's program.

Besides this general academic trend, different departments have initiated other new studies in which logic education is involved. Examples of these are artificial intelligence at the computer science department, cognitive science at the social

* Logic in Action is a project financed by the Dutch Ministry of Science and Education (the so-called Beta Platform program) and is organized by the Center for Creation, Content and Technology (http://ccct.uva.nl/content/project/beta). Many other persons are involved in the project than only the two authors of this paper. The project has been initiated by Johan van Benthem (University of Amsterdam), Jan van Eijck (Center for Mathematics and Computer Sciene Amsterdam) and Hans van Ditmarsch (University of Seville).

P. Blackburn et al. (Eds.): TICTTL 2011, LNAI 6680, pp. 141–153, 2011.

science faculty and computational linguistics at the department of humanities. Altogether, these relatively new programs meant a considerable increase of the local undergraduate consumption of logic, since in earlier days logic was only taught on an undergraduate level to students of mathematics, computer science and philosophy.

Although these academic tendencies are very beneficial for the spreading of the education of logic, there is a much more important 'internal' reason for the more dominant position of logic in academia. The development of logic, mainly caused by its influence and applications in computer science since the fifties, has changed its academic presentation dramatically. Whereas traditionally logic was taught as the abstract 'meta-science of valid reasoning' at mathematics and philosophy departments, modern logic education today focusses very much on the mathematical and computational methodology it offers for a wider range of studies. Every academic field that studies representation and exchange of information in one way or another, either by machines or real humans, requires logic as one of the fundamental mathematical tools.

1.1 The Logic in Action initiative

Logic in Action is an educational initiative which started at the University of Amsterdam, in cooperation with the Tsinghua University (Beijing, China) and the Stanford University (California, USA), in 2009. The main objective of the project is to provide new teaching material for modern introductory logic courses for the wider academic audience mentioned here above. Besides classical logic, the project focusses on modern logics, such as dynamic and epistemic logics, which provide formal and computational methods for representation and exchange of information rather than solitary reasoning.

Logic in Action provides open courseware in the form of an extensive animated e-book, which can be read and studied on computers, but also on other modern electronic devices such as certain type of e-readers, tablet computers and even modern smartphones. The e-book consists of texts with many animated illustrations, and also diverse applications: small ones that visualize the most important concepts, and larger ones that can be used to exercise the most important methodological skills. The complete electronic manuscript has been set up in such a way that the reader — maybe 'the user' would be a more appropriate typology here — may switch from text to animations or applications as smoothly as possible (single touch or click).

This paper is meant to give a bird's-eye view on what Logic in Action has to offer for introductory courses in logic. Section 2 focusses on the contents and the educational motivations behind it. Section 3 presents the electronic extensions of the e-manuscript together with some illustrative examples. Section 4 elaborates on the near future of the Logic in Action project, which goes in three directions: specialized advanced logic courses for graduate students, the creation of open courseware for pre-academic education on secondary schools and also an additional part that we are working on at the moment: learning logic by programming, and vice versa.

2 Manuscripts

Our courseware is based on the idea that today logic should be taught as a *methodological* science which is applicable in many different areas. The ultimate tool for teaching logic is by demonstrating the tools logic has to offer for those fields of study.

What comes first in our set-up of our courseware is *logical design*. In order to study a given phenomenon, we first capture the formal structures (models) that represent it, and then a formal language which allows us to describe these structures. The modelling enables us to abstract away from details to identify the key aspects of the phenomenon. The formal language facilitates a precise description and study of these properties in a systematic way. Once models and language have been defined we are ready to introduce *logical computation*, i.e., formal inference systems. The main goal of the open courseware is to look at logic from this broader methodological perspective. Instead of presenting it as an abstract isolated science that focusses merely on truth-conditions and truth-preserving reasoning, logic is presented as a methodology that takes the 'art' of *modelling* as its starting point.

In order to achieve this goal, the manuscript is divided into three main parts. First, as an introduction of the basic logical concepts, three classical systems are presented: propositional logic, syllogistic reasoning and predicate logic. Special emphasis is put on the semantic models in which formulas of these systems are evaluated in order to underscore the use of the languages as a tool for describing structures. Then we move to more recent logical developments that shift the main focus from the notion of truth to the concept of information. The fundamental topics that this new perspective puts into light are threefold: knowledge and the way it changes (dynamic epistemic logic in a broad sense), abstract actions (dynamic logic), and the way information and actions work together in multi-agent environments (interaction). Finally, after showing how formal structures and formal languages are useful for representation, description and therefore the formalization of different phenomena, the manuscript finishes with an extensive part on three different sorts of computational methods.

The remainder of this section presents a more detailed description and motivation for each of the three parts that constitute the core of the lecture notes.

2.1 Classical Systems

Traditionally, logic has been understood as the study of valid patterns of reasoning, and many systems have been developed with the aim of formalizing what a valid inference is. The most important classical frameworks take the truth-value of a sentence to be an objective concept, completely determined by the sentence's components and completely detached from any personal considerations. This is what has made mathematics the paradigmatic example of classical logic, with precise definitions setting the basic ground, and then the formalization of sound proofs showing what follows from initial assumptions.

The fundamental logical framework is that of propositional logic. This logic defines valid patterns of reasoning with sentences whose basic components are abstract propositional variables. One of the most important features of propositional logic is that it gives formal meaning to *logical connectives* (sentential operations) interpreted as Boolean functions. The models for formulas of this language are simple *truth-value assignments.* Our student first learns about compositional semantics on the basis of these simple models, and then uses this to distinguish valid from invalid inferences.

At this point there is not much difference with a traditional first introduction to the key constituents of a logical system. Despite its simplicity, we use propositional logic to offer our student a broader modernized perspective on logic. For example, besides the inevitable truth-tabling we also use the so-called *update* definition of validity which sheds another light on logical inference. The premises of an inference are seen as pieces of information that reduce the initial uncertainty of the reasoner. When all the premises have been processed in this way, a conclusion is then defined to be valid if it does not add any new information (no further reduction of uncertainty). Another informational perspective we use on this basic level is that of truth-value assignment of a propositional formula as a *game*, or rather a *debate*, between two opposing parties: one defending its truth (the verifier) and one claiming its falsity (the falsifier).

Still, as a purely abstract formal system, propositional logic can easily be underestimated as a simple innocent logic by a starting student. Besides many puzzles that we have added to the text to bring propositional logic more to life, we focus on more serious applications and deeper mathematical understanding of the impact of propositional logic. The most manifest example is binary arithmetic as performed by logical circuits built up from the propositional logical operations. Our student learns how these circuits work and, moreover it is taught that this can all be done due to the expressive wealth of the propositional language, i.e., its functional completeness. In addition we focus on the rude complexity of propositional logic. There is a computational price (NP-completeness) to be paid for the surprising richness of basic system.

The second framework is that of syllogistic reasoning. It 'opens the box' of propositional logic by looking closer at what an atomic proposition actually expresses: very often it is not about abstract sentences, but about objects and their properties. Syllogistics focuses on simple quantification patterns, like "All P are Q" and "Some P are Q", and it provides the first steps towards the more general system of predicate logic.

On the other hand, syllogisms can also be seen as a special form of propositional reasoning, and we demonstrate how syllogistic inferences can be dealt with by information updates very much in line with the procedure we have given for propositional logic. In the case of syllogistic reasoning, universal information like "All P are Q" can be processed as the removal of the P-individuals who are not Q. As a consequence of this informational interpretation it can be shown that, due to its moderate expressiveness, the computational complexity of syllogistic reasoning is lower than that of full propositional logic. This shows our student that if a logic system is brought back to a natural but limited fragment there may also be some important computational benefits to be gained.

The strongest of the logical frameworks that are presented is that of *predicate logic*, extending syllogisms to deal not only with objects and their properties, but also with relations of arbitrary arity between them, all these combined in arbitrary forms of quantification. This is the most important system in logic today because it is a universal language for talking about formal *structures*, that is, collection of objects together with their properties and relations. Predicate logic has been used to increase precision in describing and studying structures from linguistics and philosophy to mathematics and computer science.

The language of predicate logic is much harder to get acquainted with for many beginning students. To overcome this difficulty, much attention in the text is given to translations of phrases of natural language to predicate logic making use of the more accessible quantificational patterns of syllogisms. Students are trained to write formulas first as trees to recognize the composition of such patterns, and then rewrite those in the standard linear logical form.

Another difficulty of predicate logic is that its semantics is relatively much more complex than that of propositional logic and that of syllogistics. The text therefore focusses first on the *informal* semantics of pictures and finding the predicate logic formulas to distinguish two of such pictures. Moreover, much attention has been given to the role of free variables as to make the working of variable assignments more comprehensible (see also Figure 1).

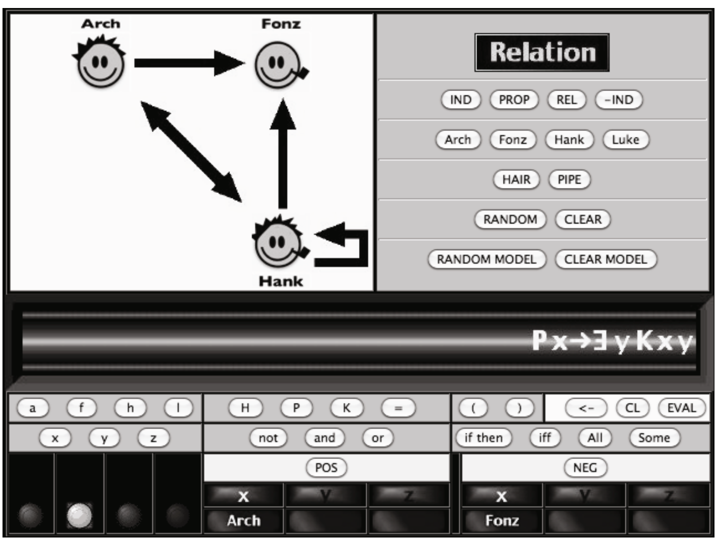

Fig. 1. The art of modeling in the context of predicate logic. This is a small application from the electronic document which illustrates model evaluation evaluation in first-order models. What is important here is that it also illustrates the working of assignments of free variables.

2.2 Knowledge, Action and Interaction

As we mentioned before, logic has been traditionally taken to be the objective study of valid reasoning, with truth-preserving inference as the most important concept and mathematical logic as the paradigmatic area. But with the emergence of artificial intelligence, the 1980s witnessed the development of many logical systems that focussed on what are nowadays called "rational agents". These proposals expanded the notion of logic: from the study of the objective notion of *truth* and the reasoning patterns that preserve it, to the study of subjective notions of *information* and the diverse actions that change it.

Three fundamental themes play a role here. The first one is the study of *information*, its diverse representations and its properties. Then, this information may provoke agents to perform *actions*, thereby changing the objective circumstances — the actual state of affairs — as well as the subjective context — the information that the agents have about the state of affairs. The third theme is *social interaction* which ties up the previous two, and enables the study of the most interesting forms of action and information flow which involves not a single but multiple agents sharing their private information by communication. For each of these main themes, the manuscript presents a logical system to formalize the most important concepts.

The first of these themes is best exemplified by epistemic logic, a logical framework that differs from classical logic in that it incorporates the reasoner(s), i.e., the agent(s), within the system. The epistemic logical language and its semantic models facilitates the representation of the private knowledge the agents have about the actual world as well as the knowledge they have about their own and other agents' knowledge.

As a simple extension of epistemic logic for modelling communication, the manuscripts outlines the so-called public announcement logic. At first, *dynamic* operators appear on stage to represent the effect of such public announcements. In addition, operators are introduced in order to express group-related notions of information, with common knowledge being the most representative. As these announcement acts are taken to be 'really public', the optimally attentive agents acquire new information, but they also get to know that the other listeners have gained the same information, and all those agents share this knowledge about their knowledge and so on (see Figure 2). The infinite nature of the common knowledge notion makes the underlying logic mathematically distinct from the classical logics of the first part of the book.[1]

In an additional chapter the manuscript focusses on the propositional dynamic logic framework, which allows us to study actions in a more abstract way. Propositional dynamic logic incorporates a more extensive combinatorics of actions and is as such a very powerful tool in many research areas. Originally the formalism has been introduced in order to develop systems to derive information about the behavior of computer programs, but nowadays it serves as a general logic of actions.

[1] For example, such extended epistemic logics lose the *compactness* property.

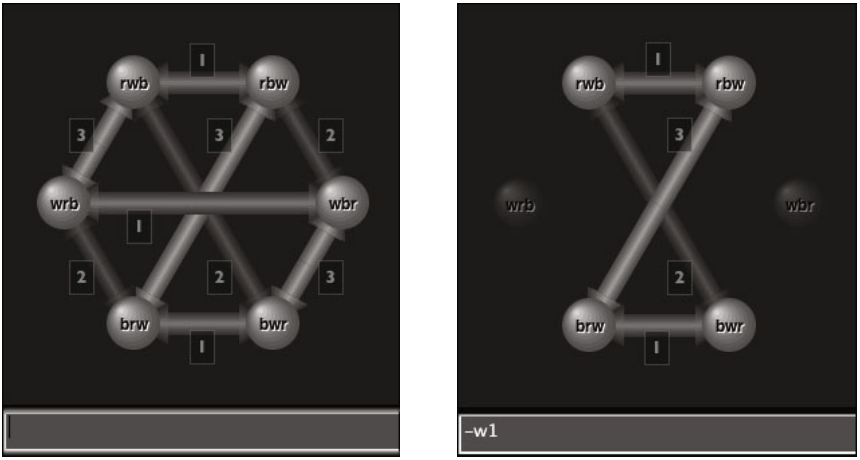

Fig. 2. A small app from the electronic document which visualizes the working of *Public Announcement Logic*. The example is about a very simple card game, three players holding a card of different colors, red, white and blue. The players can see their own card but not those of the other players. The program takes new input as public announcements, and administrates the effect as elimination of possibilities. In the second picture the first player just told (publicly) that he does not have the white card.

The third topic combines the previous two in a multi-agent environment, and deals with the broadest conception of logic nowadays: the study of agents, their private and shared attitudes towards information (knowledge, beliefs, preferences, intentions, desires, etc.), and the way these attitudes change due to the interaction between those agents. A paradigmatic application of this new understanding of logic is the analysis of competitive multi-agent situations, commonly known as *games*. Games have always played an important role in the history of logic, witness the many sorts of interactive games with two opposing players that have been used as tools for demonstrating the validity or invalidity of inferences, or for determining the truth or falsity of arguments in a dialogue. But this broader conception of logic shows how the relation works in both ways: not only games are vehicles for defining logical concepts, but logic also provides tools for analyzing games, focussing not only on their structure (the actions the players may undertake) and their payoffs (the agent's *preferences*), but also on the information agents have about each other and the way this is affected by the different actions that take place.

For the Logic in Action project this part of the book is the most challenging. Although we introduce all the aforementioned topics on an elementary level, our student becomes familiar with recent developments of logic and, at the same time, also learns about the new and central position that logic has taken up in the era of information and communication.

2.3 Methods

The final part of the lecture notes elaborately introduces three different formal systems for computing the validity of logical inferences. We start off with the tableau method since it fits best with the model-theoretic perspective of the first two parts of the book. Moreover, it is the most general method of the systems discussed in this part, because it is a testing method. Invalidity of a given inference can be determined by means of the construction of a counter-model. Validity of such an inference, on the other hand, is then shown by proving the non-existence of such counter-models. An additional important didactic feature of the tableau method is that it shows quite straightforwardly how reasoning can be implemented within computer programs.

On the other hand, a clear disadvantage of the tableau method is that it is a refutational method: instead of showing directly the truth of a given conclusion under certain circumstances a tableau demonstrates that it can not be false. An obvious consequence is that a tableau computation does not always reflect "the way people think". In the second chapter we therefore outline the method of natural deduction as a more 'human' system. The reader is taught how characteristic complete systems of axioms as introduced in the first two parts of the book can be uplifted to natural systems of reasoning by facilitating conditional reasoning by means of temporary hypotheses.

The third and final chapter of this part of the book is completely dedicated to first-order theorem proving by means of resolution and unification. This is probably the most technical chapter of the Logic in Action manuscript as it stands now, and may therefore not be suitable for every introductory course in logic. Despite the technical nature of this topic, we have chosen to incorporate this formal method since it exemplifies the practical nature of a logical formalism invented by philosophers, like Frege and Peirce, more than a hundred years ago. Today a student may as well learn predicate logic as a very useful programming language as this chapter illustrates.

3 Electronic Support

In order to provide a better understanding of the material described in the previous section, the e-book provides digital support for the users (students and teachers) in the form of animated illustrations and different sorts of applications. The former are meant as step-by-step — or rather click-by-click — visualizations of the most important concepts, and the latter are mainly meant for training logical skills. These two electronic extensions of the document are conceived as tools for the student to master technical methods, but they can also be used by teachers for demonstration purposes.

In addition to this technical support, the full electronic version of the book contains optional teaching material that relates the main logic topics with other sciences. Since the manuscript focusses on the technical methodology that formal logic offers, a student may easily lose track of the meaning of the different forms of symbolic manipulation that he is learning. In order to put these methodological developments in a proper context, the text contains several 'outlooks' that highlight important applications of the presented material. These outlooks allow the students to see the subject from different perspectives, suitable for different backgrounds and different objectives.

In the remainder of this section we will elaborate a bit further on the electronic part of our courseware.

3.1 Animated Illustrations

Visualization is an important tool for learning formal logic. Animated illustrations provide an important enhancement of our means to picture examples of technical concepts. Firstly, animations give us the possibility of building illustration in a step-by-step fashion, therefore allowing us to provide, at each stage, a proper explanation. Moreover, the examples can be replaced by other examples, either chosen by the reader himself or generated automatically.[2] In Figure 3 we have given an example of an animated illustration from the second part of the book. The reader can freely change such examples.

3.2 Applications

Besides animations the book also contains larger applications which assign a more (inter)active role to the reader. We have built several interactive applications to make it easier for the user to exercise technical skills such as model checking, model construction and setting up derivations in formal deduction systems such as axiomatic systems, semantic tableaus, natural deduction and resolution. This sort of applications have clear advantages for exercises which involve logical derivations and computations. Editing such formal structures becomes much easier and the programs check the results. This is an advantage since, in ordinary books, the provided solutions may be misleading to the reader: his own answers may be correct as well. In Figure 4 two applications are shown as examples of applications for logical derivations.[3]

3.3 Additional Contents

Besides interactive extensions, the e-book also contains other additional material. The main purpose of these additions is to offer a broader perspective on logic for specific target groups of students. They consist of applications of logic, deeper theoretical results for the methodologies as unfolded in the main text, the philosophical and historical backgrounds of the development of logic and external links to other online documents.

[2] The final electronic version of the e-book is written in HTML (5), and the animated illustrations are coded in JavaScript such that these animations run within the electronic text.

[3] A minor technical disadvantage of these larger applications is that they have to be run in a separate window of the document browser.

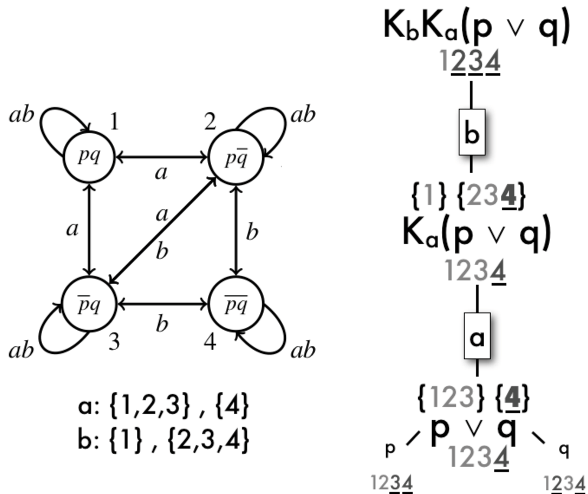

Fig. 3. This is a screen dump of how the final stage of an animation of Kripke (possible world) models are defined and how evaluation of modal formulas take place. As we found out in try-outs of our courseware students find it hard to capture all the formal details of the definition of such models. This animation generates new models on the spot for limitless training by examples. The evaluation of the formulas are built up step-by-step as to understand the compositional aspect of evaluation.

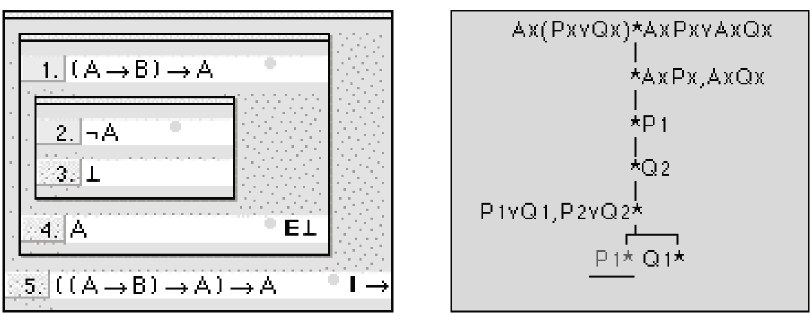

Fig. 4. Two important applications for methodological training. On the left the reader is on his way to prove Peirce's law by natural deduction (Fitch style). The editor for building the proofs facilitates bridging gaps in the proof and the program checks the soundness of every step made. On the right hand our reader is decomposing a tableau (Beth style) to test the validity of a chosen inference.

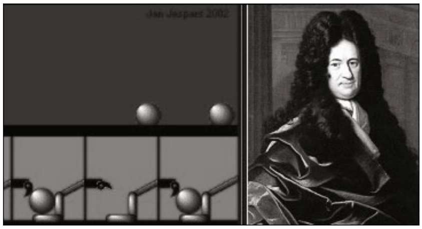

Fig. 5. A fragment of an animation of Leibniz' 'mechanica dyadica', the first binary computer (1705). This animation comes from a bibliographical outlook in our first chapter on propositional logic where the relation with binary computation is explained to our readers.

Outlooks. The most important additions in the e-book are outlooks which reflect on the applications of logic today. Such perspectives are very important to motivate the study of logic. Because a specific application may be appealing to just a selection of our potential readers, we have chosen to put these parts in the e-book version as optional extensions of the manuscript. The full electronic document is organized in such a way that the manuscript can be extended smoothly with these outlook sections.

In addition to applications of logic we have also added some outlooks on theoretical issues which are introduced on an informal level. They are meant as eye-openers for the reader, but as the course is on an elementary level these subjects are not dealt with in full technical detail. Currently we started working on a fourth part of the manuscript for graduate students for logical meta-theory (see also Section 4).

The historical and philosophical context of logic. The second group of optional extensions of our manuscripts focusses on the history and the philosophical context of logic. They provide basic background information for students of logic courses, allowing them to reach a deeper understanding and appreciation of the logical methodology. As an example see Figure 5 which shows an animated version of the first binary computing device. Although this invention has only been described on paper by its inventor Leibniz (1705), in the e-book version of our manuscript the reader can make it run.

External resources. As a program of open courseware we also benefit from many other world-wide educational initiatives of this kind. The e-book includes many references to other external resources for the teaching of logic.

Logic in Action is free and open courseware, and will therefore continuously be extended and updated. Especially the electronic part of the book will be extended considerably in the near future. But there is more what we will be working on to extend the scope of the Logic in Action project. In the next section we will outline this in a brief prospectus.

4 Logic in Action Today and in the Future

There are three major directions that the Logic in Action group is currently working on: practical logic training for students of computer science and related subjects such as artificial intelligence and computational linguistics, a full new part on the meta-theory of logic for graduate courses, and finally a series of e-books in the Logic in Action style for pre-academic education for secondary schools.

4.1 Logic in Action at Work

A particularly important development that Logic in Action is working on at the moment is a technical appendix that provides instructive material for learning how to build tools of logic. This additional part is specially meant for students in the field of computer science. The functional programming language `Haskell` is used to instruct students how they can build relatively small applications to let the logic tools really do the work.[4] In this part we will specifically aim at applications for the logics that are taught in the second part of the manuscript: epistemic logics and dynamic extensions of those, which are particularly interesting for the aforementioned group of students.[5]

At this moment this part of the course is incorporated as a technical annex, but in the future it will take a more dominant place within the Logic in Action project. For a large group of students today this type of 'hands on'-training works much better than theoretical education. Another obvious advantage is that the students learn logic and computer programming at the same time.

4.2 Graduate Courses

Another part that we are working on is an extension of our courseware meant for more advanced graduate and post-academic courses in logic. For students of mathematics and theoretical computer science we will add chapters about the meta-theory of logic. The most important themes will be completeness and incompleteness results, complexity theory and more advanced model-theory. Other topics which are planned to be incorporated are non-classical logics such as intuitionistic logic, non-monotonic logic and extended modal logics.

4.3 Pre-academic Education

In secondary schools, especially those for pre-academic education, the mathematics curricula have been extended considerably in the last fifteen years. One important reason for this change in basic pre-academic education of mathematics is the role of computer science. Whereas classical mathematical training aims at the application of mathematics within the traditional natural sciences such as physics, chemistry and biology, today pupils in Dutch secondary schools can

[4] For a textbook which shows the use of `Haskell` in this context see[2].

[5] What we have in mind here is very much in line van Eijck's `DEMO`- system [3].

choose to extend this with alternative mathematical programs for other scientific fields in which computer science and related subjects take a dominant place.

In an earlier logic education project at the University of Amsterdam, which was called 'The Dissemination of Logic' which made part of van Benthem's earlier Spinoza Project (1997–2000),[6] we have published several logic textbooks for these extended programs in Dutch secondary schools [6].[7] Logic in Action is currently working on supplementary electronic editions of these textbooks in English.

5 Conclusions

The open courseware project Logic in Action develops modern course material for the teaching of logic to students with different academic backgrounds. In the Age of information and communication formal logic is one of the most important mathematical tools that these new generations will need. Logic in Action contributes to this in two respects: in content as well as in didactics. In addition to the ordinary classical standards of a course in logic, Logic in Action provides elementary undergraduate teaching of modern logics for modelling communication. As for the didactic part, Logic in Action makes use of animated e-books to facilitate the teaching of logic.

Acknowledgements. We like to thank Johan van Benthem, Hans van Ditmarsch and Jan van Eijck who initiated the Logic in Action project. We also want say thanks to Dora Achourioti, Cédric Dégremont, Nina Gierasimczuk, Tomohiro Hoshi, Lena Kurzen, Fenrong Liu and Stefan Minica who have been using the earlier versions of the Logic in Action manuscript in their logic courses during the last two years and have provided very valuable comments and suggestions on the basis of their teaching experience with Logic in Action.

References

1. van Benthem, J., van Ditmarsch, H., van Eijck, J.: Logica in Actie. Spinoza-reeks Open Universiteit Heerlen. Academic Service, Den Haag (2009)
2. Doets, K., van Eijck, J.: The Haskell Road to Logic, Maths and Programming. Texts in Computing, vol. 4. King's College Publications, London (2004)
3. van Eijck, J.: Demo — a demo of epistemic modelling. In: van Benthem, J., Gabbay, D., Löwe, B. (eds.) Interactive Logic — Procs. of The Seventh Augustus de Morgan Workshop. Texts in Logic and Games, vol. 1, pp. 305–363 (2007)
4. van Eijck, J., Jaspars, J., Ketting, J., Pauly, M.: Denkende Machines: Computers, Rekenen en Redeneren. Exact in Context, vol. 1. Amsterdam University Press, Amsterdam (2001)
5. van Eijck, J., Visser, A.: Bewijzen en Inzien. Exact in Context, vol. 2. Amsterdam University Press, Amsterdam (2007)
6. Jaspars, J.: The dissemination of logic in dutch secondary schools. In: Procs. of The First Conference on Tools for Teaching Logic, Salamanca (2000)

[6] See `http://www.illc.uva.nl/lia/`

[7] For those who read Dutch: See the booklets [4], [5] and [1].