Lecture Notes in Computer Science 13468

FoLLI Publications on Logic, Language and Information

Subline of Lectures Notes in Computer Science

More information about this series at https://link.springer.com/bookseries/558

Agata Ciabattoni · Elaine Pimentel ·
Ruy J.G.B. de Queiroz (Eds.)

Logic, Language, Information, and Computation

28th International Workshop, WoLLIC 2022
Iași, Romania, September 20–23, 2022
Proceedings

Springer

Editors
Agata Ciabattoni
Technische Universität Wien
Vienna, Austria

Elaine Pimentel
University College London
London, UK

Ruy J.G.B. de Queiroz
Universidade Federal de Pernambuco
Recife, Pernambuco, Brazil

ISSN 0302-9743 ISSN 1611-3349 (electronic)
Lecture Notes in Computer Science
ISBN 978-3-031-15297-9 ISBN 978-3-031-15298-6 (eBook)
https://doi.org/10.1007/978-3-031-15298-6

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

This volume contains the papers presented at the 28th Workshop on Logic, Language, Information and Computation (WoLLIC 2022) held during September 20–23, 2022 at the Faculty of Computer Science, Alexandru Ioan Cuza University in Iasi, Romania. The WoLLIC series of workshops started in 1994 with the aim of fostering interdisciplinary research in pure and applied logic. The idea is to have a forum which is large enough in terms of the number of possible interactions between logic and the sciences related to information and computation, and yet small enough to allow for concrete and useful interaction among participants.

For WOLLIC 2022 there were 46 submissions. Each submission was reviewed by at least two Program Committee members. The committee decided to accept 25 papers. This volume includes all the accepted papers, together with the abstracts of the invited speakers at WOLLIC 2022:

- Anupam Das (University of Birmingham, UK),
- John Horty (University of Maryland, USA),
- Marie Kerjean (Université Sorbonne Paris Nord/CNRS, France),
- Dorel Lucanu (Alexandru Ioan Cuza University, Romania), and
- Francesca Poggiolesi (Université Sorbonne/CNRS/IHPST, France).

It also includes abstracts of the invited tutorials given by

- Gabriel Ciobanu (Romanian Academy, Romania),
- Sonia Marin (University of Birmingham, UK), and
- Marija Slavkovik (University of Bergen, Norway).

We would like to thank all the people who contributed to making WOLLIC 2022 a success. We thank the Program Committee and all additional reviewers for the work they put into reviewing the submissions. We thank the invited speakers and the tutorial presenters for their inspiring talks, the Steering Committee and the Advisory Committee for their advice, and the Local Organizing Committee members (especially Ștefan Ciobâcă) for their great support. Finally, we thank all the authors for their excellent contributions.

The help provided by the EasyChair system created by Andrei Voronkov is gratefully acknowledged. We also would like to acknowledge the scientific sponsorship of the following organizations: the Interest Group in Pure and Applied Logics (IGPL), the Association for Logic, Language and Information (FoLLI), the Association for Symbolic Logic (ASL), the European Association for Theoretical Computer Science

(EATCS), the European Association for Computer Science Logic (EACSL), and the Brazilian Logic Society (SBL).

September 2022

Agata Ciabattoni
Elaine Pimentel
Ruy de Queiroz

Organization

Program Committee Chairs

Agata Ciabattoni	Technische Universität Wien, Austria
Elaine Pimentel	University College London, UK

Steering Committee

Samson Abramsky	University College London, UK
Anuj Dawar	University of Cambridge, UK
Juliette Kennedy	University of Helsinki, Finland
Ulrich Kohlenbach	Technische Universität Darmstadt, Germany
Daniel Leivant	Indiana University Bloomington, USA
Leonid Libkin	University of Edinburgh, UK
Lawrence Moss	Indiana University Bloomington, USA
Luke Ong	University of Oxford, UK
Valeria de Paiva	Topos Institute/PUC-Rio, USA/Brazil
Ruy de Queiroz	Universidade Federal de Pernambuco, Brazil
Alexandra Silva	Cornell University, USA
Renata Wassermann	Universidade de São Paulo, Brazil

Program Committee

Arthur Azevedo de Amorim	Boston University, USA
Diana Costa	University of Lisbon, Portugal
Hans van Ditmarsch	Open Universiteit, The Netherlands
Rajeev Goré	Polish Academy of Sciences, Poland
Roman Kuznets	Technische Universität Wien, Austria
João Marcos	Federal University of Rio Grande do Norte, Brazil
Lawrence Moss	Indiana University Bloomington, USA
Valeria de Paiva	Topos Institute/PUC-Rio, USA/Brazil
Revantha Ramanayake	University of Groningen, The Netherlands
Mehrnoosh Sadrzadeh	University College London, UK
Alexandra Silva	Cornell University, USA
Alex Simpson	University of Ljubljana, Slovenia
Sonja Smets	University of Amsterdam, The Netherlands
Alwen Tiu	Australian National University, Australia
Leon van der Torre	University of Luxembourg, Luxembourg
Andrea Aler Tubella	Umeå University, Sweden
Andrés Villaveces	Universidad Nacional de Colombia, Colombia
Renata Wassermann	University of São Paulo, Brazil

Additional Reviewers

Henrique Antunes	Federal University of Bahia, Brazil
James Brotherston	University College London, UK
Francicleber M. Ferreira	Federal University of Ceará, Brazil
Daniel Figueiredo	University of Aveiro, Portugal
Robert Freiman	Technische Universität Wien, Austria
Jakub Gajarský	University of Warsaw, Poland
Wesley Holliday	University of California, Berkeley, USA
Peter Jipsen	Chapman University, USA
Louwe B. Kuijer	University of Liverpool, UK
Timo Lang	University College London, UK
Giacomo Lenzi	University of Salerno, Italy
Xavier Parent	Technische Universität Wien, Austria
Todd Schmid	University College London, UK
Stephan Schulz	DHBW Stuttgart, Germany
Ian Shillito	Australian National University, Australia
Julian Sutherland	Imperial College London, UK
Sebastiaan Terwijn	Radboud University Nijmegen, The Netherlands
Apostolos Tzimoulis	Vrije Universiteit Amsterdam, The Netherlands
Daniel Ventura	University of Goiás, Brazil
Amanda Vidal	Spanish National Research Council, Spain

Invited Talks

On the Proof Theoretic Strength of Circular Reasoning

Anupam Das

University of Birmingham, UK
a.das@bham.ac.uk

Cyclic and *non-wellfounded* proofs are now a common technique for demonstrating metalogical properties of systems incorporating (co)induction, including modal logics, predicate logics, type systems and algebras. Unlike usual proofs, non-wellfounded proofs may have infinite branches: they are generated *coinductively* from a set of inference rules. Naturally, such 'proofs' may admit fallacious reasoning, and so one typically employs some global correctness condition inspired by ω-automaton theory.

A key motivation in cyclic proof theory is the so-called 'Brotherston-Simpson conjecture': are cyclic proofs and inductive proofs equally powerful? Naturally, the answer depends on how one interprets 'equally powerful', e.g. as provability, proof complexity, logical complexity etc., as well as on the logic at hand. In any case it is interesting to note that the tools employed in cyclic proof theory are often bespoke to the underlying logic, yielding a now myriad of techniques at the interface between several branches of mathematical and computational logic.

In this talk I will discuss a line of work that attempts to understand the expressivity of cyclic proofs via forms of proof theoretic strength. Namely, I address predicate logic in the guise of first-order arithmetic, and type systems in the guise of higher-order primitive recursion, and establish a recurring theme: circular reasoning buys precisely one level of 'abstraction' over inductive reasoning. Along the way we shall see some of the aforementioned interplays in action, in particular exploiting techniques from proof theory, reverse mathematics, automaton theory, metamathematics, rewriting theory and higher-order computability.

Open Texture and Defeasible Semantic Constraint

John Horty

University of Maryland, USA
horty@umd.edu

The concept of open-texture was introduced in [8], with its importance for legal theory noted shortly afterward in [2]. Due to the intrinsic interest and practical importance of the issues surrounding open-textured predicates, a substantial literature on the topic has evolved within legal theory; some highlights include [1, 5–7]. For the most part, however, this literature focuses on broader issues in the theory of open texture—the role of defeasible legal rules, policy arguments concerning the application of these rules, the impact of open-textured predicates on theories of legal interpretation, connections with philosophy of language very generally. The literature does not provide anything like a semantic theory of open-textured predicates. In this talk, I will attempt to supply such a theory.

The talk has four parts. In the first, I will review some of the problems presented by open-textured predicates, and suggest an explication of the concept according to which: the predicate P is open-textured just in case, given any description of an object a on the basis of which we can reasonably apply P to a, it is always possible consistently to extend this description in such a way that it is no longer reasonable to apply P to a. In the second, I will sketch an account of constraint in the common law presented in my own recent work [3, 4]. In the third part, I will show how this account can be adapted to help us understand open-textured predicates as well. Finally, in the fourth part, I will talk a bit about the reasoning involved in reaching decisions that satisfy the account of constraint, and show how this reasoning can be modeled in a simple defeasible logic.

References

1. Baker, G.: Defeasibility and meaning. In: Hacker, P., Raz, J. (eds.) Law, Morality, and Society: Essays in Honour of H. L. A Hart, pp. 26–57. Oxford University Press (1977)
2. Hart, H.L.A.: Positivism and the separation of law and morals. Harvard Law Rev. **71**, 593–629 (1958)
3. Horty, J.: Rules and reasons in the theory of precedent. Legal Theor. **17**, 1–33 (2011)
4. Horty, J.: Constraint and freedom in the common law. Philosopher's Imprint **15**(25) (2015)
5. MacCormick, N.: On open texture in law. In: Amslek, P., MacCormick, N. (eds.) Controversies about Law's Ontology, pp. 72–83. Edinburgh University Press (1991)

6. Schauer, F.: A critical guide to vehicles in the park. New York University Law Review, **83**, 1109–1134 (2008)
7. Tur, R.: Defeasibilism. Oxford J. Legal Stud. **21**, 355–368 (2001)
8. Waismann, F.: Verifiability. In: Proceedings of the Aristotelian Society, Supplementary. vol. 19, pp. 119–150. Harrison and Son (1945), reprinted in Anthony Flew (eds.) Logic and Language, pp. 117–144, Blackwell Publishing Company, 1951; pagination refers to this version.

∂ is for Dialectica

Marie Kerjean[1,2] and Pierre-Marie Pédrot[3]

[1] CNRS
[2] Université Sorbonne Paris Nord
[3] Inria, France
kerjean@lipn.fr,pedrot@inria.fr

Dialectica was originally introduced by Gödel in a famous paper [7] as a way to constructively interpret an extension of HA [1], but turned out to be a very fertile object of its own. Judged too complex, it was quickly simplified by Kreisel into the well-known realizability interpretation that now bears his name. Soon after the inception of Linear Logic (LL), Dialectica was shown to factorize through Girard's embedding of LJ into LL, purveying an expressive technique to build categorical models of LL [13]. In its logical outfit, Dialectica led to numerous applications and was tweaked into an unending array of variations in the proof mining community [10].

The modern way to look at Dialectica is however to consider it as a program translation, or more precisely *two* mutually defined translations of the λ -calculus exposing intensional information [14].

In a different scientific universe, Automatic Differentiation [8] (AD) is the field that studies the design and implementation of *efficient* algorithms computing the differentiation of mathematical expression and numerical programs. Indeed, due to the chain rule, computing the differential of a sequence of expressions involves a choice, namely when to compute the value of a given expression and when to compute the value of its derivative. Two extremal algorithms coexist. On the one hand, forward differentiation [16] computes functions and their derivatives pairwise in the order they are provided, while on the other hand reverse differentiation [12] computes all functions first and then their derivative in reverse order. Depending on the setting, one can behave more efficiently than the other. Notably, reverse differentiation has been critically used in the fashionable context of deep learning.

Differentiable programming is a rather new and lively research domain aiming at expressing automatic differentiation techniques through the prism of the traditional tools of the programming language theory community. As such, it has been studied through continuations [15], functoriality [6], and linear types [4]. It led to a myriad of implementation over rich programming languages, proven correct through semantics of higher-order differentiable functions [11]. Surprisingly, these various principled explorations of automatic differentiation are what allows us to draw a link between Dialectica and differentiation in logic.

The simple, albeit fundamental claim of this talk is that, behind its different logical avatars, the Dialectica translation is in fact a reverse differentiation algorithm, where the linearity and involutivity of differentiation have been forgotten. In the domain of proof theory, differentiation has been very much studied from the point of view of *linear logic*. This led to Differential Linear Logic [5] (DiLL), differential categories [3], or the differential λ-calculus. To support our thesis with evidence, we will formally state a correspondence between each of these objects and the corresponding Dialectica interpretation.

More generally, Dialectica is known for extracting *quantitative* information from proofs [10], and this relates very much with the quantitative point of view that differentiation has brought to λ-calculus [2]. Herbelin also notices at the end of its paper realizing Markov's rule through delimited continuations that this axiom has the type of a differentiation operator [9]. If time permits, we will explore the possible consequences of formally relating reverse differentiation and Dialectica to proof mining and Herbelin's work in the conclusion.

References

1. Avigad, J., Feferman, S.: Gödel's functional ('dialectica') interpretation. In: Buss, S.R. (eds.) Handbook of Proof Theory. Elsevier Science Publishers (1998)
2. Barbarossa, D., Manzonetto, G.: Taylor subsumes scott, berry, kahn and plotkin. Proc. ACM Program. Lang. **4**(POPL), 1:1–1:23 (2020)
3. Blute, R.F., Cockett, J.R.B., Seely, R.A.G.: Differential categories. Math. Structures Comput. Sci. **16**(6) (2006)
4. Brunel, A., Mazza, D., Pagani, M.: Backpropagation in the simply typed lambda-calculus with linear negation. POPL (2020)
5. Ehrhard, T., Regnier, L.: Differential interaction nets. Theor. Comput. Sci. **364**(2) (2006)
6. Elliott, C.: The simple essence of automatic differentiation. Proc. ACM Program. Lang. (ICFP) (2018)
7. Gödel, K.: Über eine bisher noch nicht benützte Erweiterung des finiten Standpunktes. Dialectica **12**, 280–287 (1958)
8. Griewank, A., Walther, A.: Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation. Society for Industrial and Applied Mathematics, USA, second edn. (2008)
9. Herbelin, H.: An intuitionistic logic that proves markov's principle. LICS (2010)
10. Kohlenbach, U.: Applied Proof Theory - Proof Interpretations and their Use in Mathematics. Springer Monographs in Mathematics, Springer, Heidelberg (2008). DOI: https://doi.org/10.1007/978-3-540-77533-1
11. Krawiec, F., Peyton Jones, S., Krishnaswami, N., Ellis, T., Eisenberg, R.A., Fitzgibbon, A.: Provably correct, asymptotically efficient, higher-order reverse-mode automatic differentiation. Proc. ACM Program. Lang. **6**(POPL) (jan 2022)
12. Linnainmaa, S.: Taylor expansion of the accumulated rounding error. BIT Numerical Mathematics **16**, 146–160 (1976)
13. de Paiva, V.: A dialectica-like model of linear logic. In: Category Theory and Computer Science (1989)
14. Pédrot, P.: A functional functional interpretation. CSL-LICS '14, Vienna, Austria, July 14–18 (2014)
15. Wang, F., Zheng, D., Decker, J., Wu, X., Essertel, G.M., Rompf, T.: Demystifying differentiable programming: Shift/reset the penultimate backpropagator. **3**(ICFP) (2019)
16. Wengert, R.E.: A simple automatic derivative evaluation program. Commun. ACM **7**(8), 463–464 (1964)

How to Define Domain Specific Logics using Matching Logic

Dorel Lucanu

Alexandru Ioan Cuza University, Iaşi, Romania
dorel.lucanu@gmail.com

Matching logic [2–4] is a logic that allows to uniformly specify and reason about programming languages and properties of their programs. The syntax of matching logic is simple and compact:

$$\varphi ::= x \mid X \mid \sigma \mid \varphi_1 \varphi_2 \mid \bot \mid \varphi_1 \rightarrow \varphi_2 \mid \exists \cdot x\varphi \mid \mu X \cdot \varphi$$

These eigth syntax constructs build matching logic formulas, called *patterns*, which, semantically speaking, can be matched by a set of elements. Patterns can match structures that are of certain shapes, satisfy certain dynamic properties, or meet certain logical constraints, usually all of these together.

The matching logic is endowed with a proof system that defines the provability relation, written $\Gamma \vdash_{ML} \varphi$, which means that φ is formally derivable from the axioms in Γ, using the matching logic (Hilbert-style) proof system [2].

Many important logics and/or formal systems have been shown to be definable in matching logic as logical theories. In this we consider a different approach: starting from a matching logic theory specifying a domain D, we derive a logic (proof system) $\vdash_D$ that can be used independently to reason within D.

Next we present two matching logic theories: DEF and NAT. DEF introduces a new symbol def, called the *definedness* symbol, and defines the (Definedness) axiom. This symbol and its axioms is all it is needed to define predicates, its possible values being $\bot$ or $T \equiv \neg \bot$. Then, the equality, the inclusion, and the membership are introduced as notations for patterns using the new symbol.

theory DEF
 Symbols: def
 Notations: $\lceil \varphi \rceil \equiv \mathsf{def}\ \varphi$
 Axioms: (Definedness) $\forall x.\, \lceil x \rceil$
 Notations:
 (totality) $\lfloor \varphi \rfloor \equiv \neg \lceil \neg \varphi \rceil$
 (equality) $\varphi_1 = \varphi_2 \equiv \lfloor \varphi_1 \leftrightarrow \varphi_2 \rfloor$
 (inlclusion) $\varphi_1 \subseteq \varphi_2 \equiv \lfloor \varphi_1 \rightarrow \varphi_2 \rfloor$
 (membership) $x \in \varphi \equiv x \subseteq \varphi$
endtheory

theory NAT Imports: DEF
 Symbols: $\mathbb{N}, \mathit{zero}, \mathit{succ}$
 Notations:
 $0 \equiv \mathit{zero}, 1 \equiv \mathit{succ}\, 0, 2 \equiv \mathit{succ}\, 1, \ldots$
 $\forall x{:}\mathbb{N}.\, \varphi \equiv x \in \mathbb{N} \rightarrow \varphi$
 $\exists x{:}\mathbb{N}.\, \varphi \equiv x \in \mathbb{N} \wedge \varphi$
 Axioms:
 (Zero) $\exists x{:}\mathbb{N}.\, \mathit{zero} = x$
 (Succ) $\forall x{:}\mathbb{N}.\, \exists y{:}\mathbb{N}.\, \mathit{succ}\ x = y$
 (Succ.1) $\mathit{succ}\ \mathit{zero} \neq \mathit{zero}$
 (Succ.2) $\forall x{:}\mathbb{N}.\, \forall y{:}\mathbb{N}.\, \mathit{succ}\, x = \mathit{succ}\, y \rightarrow x{=}y$
 (Domain) $\mathbb{N} = \mu D.\, \mathit{zero} \vee \mathit{succ}\ D$
endtheory

The theory NAT specifies the natural numbers up to an isomorphism [1]. Note the 1–1 correspondence between the NAT axioms and the Peano axioms (see, e.g., https://www.britannica.com/science/Peano-axioms).

From the theory DEF we may derive the following the following inference system that can be used to reason about the equality and the membership:

$$\frac{}{\vdash_{\mathsf{DEF}} \varphi = \varphi} \qquad \frac{}{\vdash_{\mathsf{DEF}} \varphi_1 = \varphi_2 \wedge \psi[\varphi_1/x] \to \psi[\varphi_2/x]}$$

$$\frac{\vdash_{\mathsf{DEF}} \varphi}{\vdash_{\mathsf{DEF}} \forall x. x \in \varphi} x \notin FV(\varphi) \qquad \frac{\vdash_{\mathsf{DEF}} \forall x. x \in \varphi}{\vdash_{\mathsf{DEF}} \varphi} x \notin FV(\varphi)$$

$$\frac{}{\vdash_{\mathsf{DEF}} x \in y = (x = y)} \qquad \frac{}{\vdash_{\mathsf{DEF}} x \in \neg\varphi = \neg(x \in \varphi)}$$

$$\frac{}{\vdash_{\mathsf{DEF}} (x \in \varphi_1 \wedge \varphi_2) = (x \in \varphi_1) \wedge (x \in \varphi_2)} \qquad \frac{}{\vdash_{\mathsf{DEF}} (x \in \exists y. \varphi) = \exists y. (x \in \varphi)}$$

$$\frac{\vdash_{\mathsf{DEF}} \varphi_1 = \varphi_2}{\vdash_{\mathsf{DEF}} \varphi_2 = \varphi_1} \qquad \frac{\vdash_{\mathsf{DEF}} \varphi_1 = \varphi_2 \quad \vdash_{\mathsf{DEF}} \varphi_2 = \varphi_3}{\vdash_{\mathsf{DEF}} \varphi_1 = \varphi_3}$$

The derived inference system for NAT imports $\vdash_{\mathrm{DEF}}$ (the first rule), includes the axioms of NAT as rules (the next four rules), and rules for inductive reasoning (the last three rules), obtained using the (PreFixPoint) and (Knaster-Tarski) from the matching logic proof system [2]:

$$\frac{\vdash_{\mathsf{DEF}} \varphi}{\vdash_{\mathsf{NAT}} \varphi} \qquad \frac{}{\vdash_{\mathsf{NAT}} 0 \in \mathbb{N}} \qquad \frac{\vdash_{\mathsf{NAT}} \varphi \in \mathbb{N}}{\mathsf{NAT} \vdash \mathit{succ}\ \varphi \in \mathbb{N}}$$

$$\frac{}{\vdash_{\mathsf{NAT}} \mathit{succ}\ 0 \neq 0} \qquad \frac{}{\vdash_{\mathsf{NAT}} \forall x{:}\mathbb{N}. \forall y{:}\mathbb{N}.\ \mathit{succ}\, x = \mathit{succ}}$$

$$\frac{\vdash_{\mathsf{NAT}} \mathit{succ}\ \varphi \to \varphi}{\vdash_{\mathsf{NAT}} \forall x{:}\mathbb{N}. (x \in \varphi \to \mathit{succ}\ x \in \varphi)} \qquad \frac{\vdash_{\mathsf{NAT}} \forall x{:}\mathbb{N}. (x \in \varphi \to \mathit{succ}\ x \in \varphi)}{\vdash_{\mathsf{NAT}} \mathit{succ}\ \varphi \to \varphi}$$

$$\frac{\vdash_{\mathsf{NAT}} \mathit{zero} \to \varphi \quad \vdash_{\mathsf{NAT}} \mathit{succ}\ \varphi \to \varphi}{\vdash_{\mathsf{NAT}} \mathbb{N} \to \varphi}$$

We obviously have $\vdash_{\mathrm{DEF}} \varphi$ implies $\mathsf{DEF} \vdash_{\mathrm{ML}} \varphi$ and $\vdash_{\mathrm{NAT}} \varphi$ implies $\mathsf{DEF} \vdash_{\mathrm{ML}} \varphi$.

We start with a gentle introduction of matching logic, including its proof system, and then we use several canonical examples of domains specified in matching logic to show how we can derive their specific logics. These examples will involve both the inductive and coinductive reasoning.

References

1. Chen, X., Lucanu, D., Roşu, G.: Initial algebra semantics in matching logic. Technical Report. University of Illinois at Urbana-Champaign (July 2020). http://hdl.handle.net/2142/107781, submitted

2. Chen, X., Lucanu, D., Roşu, G.: Matching logic explained. J. Log. Algeb. Meth. Program. **120,** 100638 (2021)
3. Chen, X., Roşu, G.: Matching mu-logic. In: Proceedings of the 34th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS 2019) (To appear) (2019)
4. Roşu, G.: Matching logic. Log. Meth. Comput. Sci. **13**(4), 1–61 (2017)

The Value of Normal Derivations in the Realm of Explanations

Francesca Poggiolesi

IHPST, UMR 8590 CNRS, Université Paris 1 Panthéon-Sorbonne

Abstract. The concept of explanation is and has long been the object of deep and wide philosophical debates; in particular it is the notion of causal explanation that has for decades dominated the general attention, e.g. see [11]. Beside the debate on causal explanation, in recent years another type of explanation has gained attention, namely *mathematical explanation*. The expression *mathematical explanations* is an umbrella term that indicates several different phenomena; in this context, we use it to refer to those mathematical proofs that not only show the theorem they prove to be true, but that they also reveal the *reasons* why the theorem it true. The idea that certain mathematical proofs have an explanatory power has been shown to be widespread amongst mathematicians (e.g. see [4]) and to have a long and illustrious philosophical pedigree (e.g. see [3] and [9]). Moreover it is a type of mathematical explanations that has been having a central role in the recent literature on the subject. *To date there has been a tendency to approach the topic of mathematical explanations by investigating the distinction between explanatory and non-explanatory proofs. This is very natural since it is widely acknowledge that some proofs are explanatory whilst other are not.* [1, p. 3]

In the attempt of better understanding mathematical explanatory proofs, some scholars have drawn an analogy with normalized derivations in natural deduction calculi, e.g. see [2, 8]. This analogy rests on a feature that both mathematical explanatory proofs and normalized derivations share, namely a complexity's increase from the assumptions to the conclusion of proofs/derivations. On the one hand, one of the main features of explanatory proofs amounts to the fact that they explain the theorem they prove by providing grounds or reasons that are simpler than the theorem they prove. On the other hand, normalized proofs typically satisfy the subformula property[1] and the subformula property can be seen as the formalization of this idea of complexity's increase from the premisses to the conclusion (e.g. see [10]). Although, for several reasons,[2] normalized derivations cannot be considered as a proper formalization of explanatory mathematical proofs, they nevertheless represent a first step towards this direction.

In this talk, the main aim is to deepen the analysis on the relationships between mathematical explanatory proofs and normal derivations; we will do that by proposing a novel model for mathematical explanations according to which when a mathematical proof is (thought of as) explanatory, then there

[1] At least under certain conditions.

[2] E.g. see [6].

exists a way to formalize it with a normal derivation where the undischarged assumptions are less complex than the conclusion. This modeling of explanatory proofs will involve the use of theorems or mathematical definitions (that occur in the mathematical proof in key positions) as rules of the derivation (e.g. see [5]), as well as the extension of the notion of logical complexity to the level of concepts (e.g. see [7]).

References

1. Baron, S., Colyvan, M., Ripley, D.: A counterfactual approach to explanation in mathematics. Philos. Math. **28,** 1–34 (2020)
2. Casari, E.: Matematica e verità. Rivista di Filosofia, **78**(3), 329–350 (1987)
3. Kitcher, P.: Explanatory unification. Philos. Science, **48**(4), 507–531 (1981) URL: http://www.jstor.org/stable/186834, doi:10.2307/186834
4. Mancosu, P.: Mathematical explanation: problems and prospects. Topoi, **20**(1), 97–117 (2001). URL: http://link.springer.com/article/10.1023/A3A1010621314372,
5. Marin, S., Miller, D., Pimentel, E., Volpe, M.: Synthetic inference rules for geometric theories. **1**, 1–28 (2021). Submitted
6. Poggiolesi, F.: On defining the notion of complete and immediate grounding. Synthese, **193,** 3147–3167 (2016)
7. Poggiolesi, F.: Explanations in mathematics: an analysis via formal proofs and conceptual complexity, 1–34 (2022). Submitted
8. Rumberg, A.: Bolzano's theory of grounding against the background of normal proofs. Rev. Symbol. Log. **6**(3), 424–459 (2013)
9. Steiner, M.: Mathematical explanation. Philos. Stud, **34**(2), 135–151 (1978). URL: http://link.springer.com/article/10.1007/BF00354494,
10. Troelstra, A.S., Schwichtenberg, H.: Basic Proof Theory. Cambridge University Press (1996)
11. Woodward, J.: Scientific explanation. In: Edward, N., Zalta, (eds.). The Stanford Encyclopedia of Philosophy (2017)

Tutorials

Cardinalities, Infinities and Choice Principles for Finitely Supported Sets

Andrei Alexandru[1] and Gabriel Ciobanu[1,2]

[1] Romanian Academy, Institute of Computer Science, Iaşi
andrei.alexandru@iit.academiaromana-is.ro
[2] Alexandru Ioan Cuza University, Iaşi, Romania
gabriel@info.uaic.ro

Finitely supported sets are standard sets equipped with actions of a group of permutations of some basic elements (atoms) whose internal structure is ignored, sets satisfying a finite support requirement. They allow a discrete (finitary) representation of possibly infinite structures containing enough symmetries to be concisely handled. The results presented in this tutorial deal with three topics:

Results Regarding Choice. The choice principles **HP** (Hausdorff maximal principle) **ZL** (Zorn lemma), **DC** (principle of dependent choice), **CC** (principle of countable choice), **PCC** (principle of partial countable choice), **AC(fin)** (axiom of choice for finite sets), **Fin** (principle of Dedekind finiteness), **PIT** (prime ideal theorem), **UFT** (ultrafilter theorem), **OP** (total ordering principle), **KW** (Kinna-Wagner selection principle), **SIP** (principle of existence of right inverses for surjections), **FPE** (finite powerset equipollence principle) and **GCH** (generalized continuum hypothesis) fail in the framework of finitely supported sets.

Results Regarding Cardinalities. Two finitely supported sets X and Y have the same cardinality (i.e., $|X| = |Y|$) if and only if there exists a finitely supported bijection $f : X \to Y$. While some arithmetic properties of cardinalities (regarding sums, products and exponents) are naturally translated from the non-atomic framework, there are also some specific atomic properties. Let us consider:

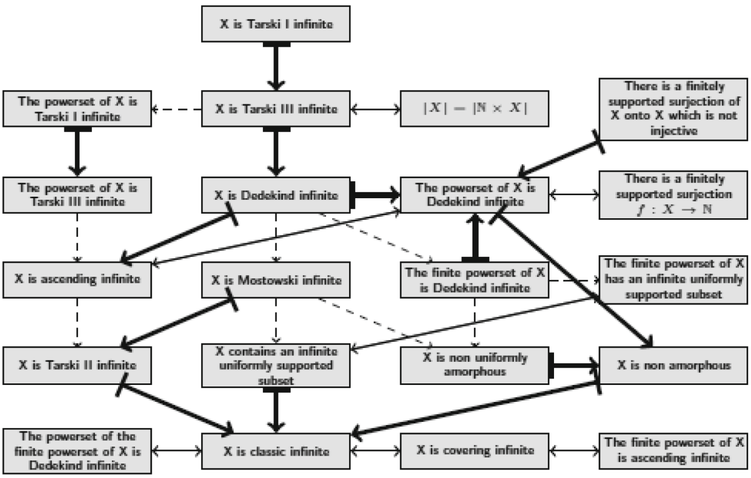
- $\leq$ given by $|X| \leq |Y|$ iff there is a finitely supported injection $f : X \to Y$;
- $\leq^*$ given by $|X| \leq^* |Y|$ iff there is a finitely supported surjection $g : Y \to X$.

We prove that the relation $\leq$ is equivariant, reflexive, anti-symmetric and transitive, but it is not total, while the relation $\leq^*$ is equivariant, reflexive and transitive, but it is not anti-symmetric, nor total.

Results Regarding Infinities. We present relationships between various pairwise non-equivalent forms of infinity defined below. Let X be a finitely supported set.

1. X is called *classic infinite* if it can be represented in the form $\{x_1, \ldots, x_n\}$.
2. X is *covering infinite* if there is a finitely supported directed family $\mathcal{F}$ of finitely supported sets with the property that X is contained in the union of the members of $\mathcal{F}$, but there does not exist $Z \in \mathcal{F}$ such that $X \subseteq Z$.
3. X is called *Tarski I infinite (TI i)* if $|X| = |X \times X|$.

4. X is called *Tarski II infinite (TII i)* if there exists a finitely supported totally ordered family of finitely supported subsets of X having no maximal element.
5. X is called *Tarski III infinite (TIII i)* if $|X| = |X + X|$.
6. X is called *Mostowski infinite (M i)* if there exists an infinite finitely supported totally ordered subset of X.
7. X is called *Dedekind infinite (D i)* if there exists a finitely supported one-to-one mapping of X onto a finitely supported proper subset of X (or equivalently, iff there exists a finitely supported one-to-one mapping $f : \mathbb{N} \to X$).
8. X is called *ascending infinite (Asc i)* if there is a finitely supported increasing countable chain of finitely supported sets $X_0 \subseteq X_1 \subseteq \ldots \subseteq X_n \subseteq \ldots$ with $X \subseteq \cup X_n$, but there does not exist $n \in \mathbb{N}$ such that $X \subseteq X_n$.
9. X is called *non-amorphous (N-am)* if X contains two disjoint, infinite, finitely supported subsets.



Relationships between several forms of infinity; the 'ultra thick arrows' indicate strict implications, the 'thin dashed arrows' indicate implications for which we did not prove yet if they are strict or not, and the 'thick bidirectional arrows' indicate equivalences.

Examples of finitely supported sets satisfying various forms of infinity:

Set	TI i	TIII i	D i	M i	Asc i	TII i	N-am
A (the set of all atoms)	No	No	No	No	No	No	No
nA, A^n, $n > 1$	No	No	No	No	No	No	Yes
$\wp_{fin}(X)$, X not D i	No	No	No	No	Yes	Yes	Yes
$X_{fs}^{A^n}$, X not D i	No	No	No	No	Yes	Yes	Yes
$A \cup \mathbb{N}$	No	No	Yes	Yes	Yes	Yes	Yes
$A \times \mathbb{N}$, $\wp_{fs}(A \cup \mathbb{N})$	No	Yes	Yes	Yes	Yes	Yes	Yes
$A_{fs}^{\mathbb{N}}$, $\mathbb{N}_{fs}^{A}$, $\wp_{fs}(\wp_{fs}(\wp_{fs}(A)))$	Yes	Yes	Yes	Yes	Yes	Yes	Yes

Reference

1. Alexandru, A., Ciobanu, G.: Foundations of finitely supported structures. A Set Theoretical Viewpoint. Springer, Cham (2020). DOI: https://doi.org/10.1007/978-3-030-52962-8

Intuitionistic Modal Proof Theory

Sonia Marin

University of Birmingham, UK
s.marin@bham.ac.uk

Intuitionistic modal logic, despite more than seventy years of investigation [4], still partly escapes our comprehension.

Already answering what is the intuitionistic variant of normal modal logic K is not obvious. Lacking De Morgan duality, there are several variants of the normal k axiom that are classically but not intuitionistically equivalent. Five axioms have been considered as primitives in the literature. An intuitionistic variant of K can then be obtained from intuitionistic propositional logic IPL by

- adding the *necessitation rule*: $\Box A$ is a theorem if A is a theorem; and
- adding a subset of the following five axioms:

$$\mathsf{k}_1: \ \Box(A \to B) \to (\Box A \to \Box B) \qquad \mathsf{k}_3: \ \Diamond(A \vee B) \to (\Diamond A \vee \Diamond B)$$
$$\mathsf{k}_2: \ \Box(A \to B) \to (\Diamond A \to \Diamond B) \qquad \mathsf{k}_4: \ (\Diamond A \to \Box B) \to \Box(A \to B)$$
$$\mathsf{k}_5: \ \Diamond\bot \to \bot$$

Structural proof theoretic accounts of intuitionistic modal logic have adopted either the paradigm of *labelled deduction* in the form of labelled natural deduction and sequent systems [6], or the one of *unlabelled deduction* in the form of sequent [2] or nested sequent systems [1, 7]. In this tutorial, we would like to give an overview of the current landscape of intuitionistic modal proof theory and illustrate how "old and new" approaches can complement each other.

We will review ordinary sequent calculi, which are adequate to treat logics based on a subset of k_1, k_2, k_3 and k_5, as well as *labelled* and *nested sequents*, which have been used to give deductive systems for the logics that cannot seem to be handled in ordinary sequent calculi, i.e., the ones that include k_4.

Both of these approaches (labelled and unlabelled) are still under active investigation. A framework for fragments of intuitionistic modal logics was recently designed, based on unlabelled sequents but related to a new intuitionistic version of *neighbourhood semantics* [3]. Another one proposes a refined labelled approach taking full advantage of the more standard *birelational semantics* [5]. As these allow for a fine-grained account of intuitionistic modal logic, we hope they will help shed some light on the intricacte world of intuitionistic modal logics.

References

1. Arisaka, R., Das, A., Strasßburger, L.: On nested sequents for constructive modal logics. Log. Methods Comput. Sci. **11**(3) (2015)
2. Bierman, G.M., de Paiva, V.: Intuitionistic necessity revisited. School of Computer Sciences-University of Birmingham (1996). Technical Report
3. Dalmonte, T., Grellois, C., Olivetti, N.: Intuitionistic non-normal modal logics: a general framework. J. Philos. Log. **49**(5), 833–882 (2020)
4. Fitch, F.B.: Intuitionistic modal logic with quantifiers **7**(2), 113–118 (1948)
5. Marin, S., Morales, M., Straßburger, L.: A fully labelled proof system for intuitionistic modal logics. J. Log. Comput. **31**(3), 998–1022 (2021)
6. Simpson, A.: The Proof Theory and Semantics of Intuitionistic Modal Logic. Ph.D. Thesis, University of Edinburgh (1994)
7. Straßburger, L.: Cut elimination in nested sequents for intuitionistic modal logics. In: Pfenning, F. (ed.) Foundations of Software Science and Computation Structures. FoSSaCS 2013. Lecture Notes in Computer Science, vol. 7794. Springer, Berlin, Heidelberg (2013). https://doi.org/10.1007/978-3-642-37075-5_14

Automating Moral Reasoning

Marija Slavkovik

University of Bergen, Norway
marija.slavkovik@uib.no

Abstract. Machine ethics has, as its topic of research, the behaviour of machines towards humans and other machines. One aspect of that research problem is enabling machines to reason about right and wrong. The automation of moral reasoning is on one end the field of dreams and speculative fiction, but on the other it is a very real need to ensure that the artificial intelligence used to automate various tasks that require intelligence does not neglect the ethical and value impact this 'replacement' of man with machine has. This tutorial introduces the problem of making moral decisions and gives a general overview of how a computational agent can be constructed to make moral decisions.

Keywords: Machine ethics · Artificial morality · Computational agency

What is Machine Ethics? Artificial intelligence (AI) is concerned with the problem of using computation to automate tasks that require intelligence [3]. In a society, we all affect each other with our activities and decisions. Ethics (or moral philosophy) is concerned with understanding and recommending right and wrong behaviours and decisions [6]. The right decisions being characterised by taking into consideration not only ones own interest, but also the interest of others [7]. The more computationally automated tasks are used to complement or replace people's tasks, the more concerns we have to ensure that the resulting actions and choices are not only correct and rational, but also do not have a negative ethical impact on society.

One way to ensure that AI has a non-negative ethical impact on society is to consider that moral reasoning is itself a cognitive task that we can consider automating. Machine ethics, or artificial morality, is a sub-field in AI that is researching this approach. The problem of automating moral reasoning can be considered as a problem of moral philosophy, whereas one is interested in questions such as: should machines be enabled with ethical reasoning [5], which norms should machines follow [8], can machines ever be moral agents [4], etc. As a problem of computer science, machine ethics focuses on the question of how to automate moral reasoning [2, 9].

Here we are concerned with the question of how to automate moral reasoning. Although this problem, and machine ethics in general, have been raised since 2006 [1], it is an extremely difficult problem that requires a lot of improvement in the state of the art in AI and moral philosophy. We discuss the basic approaches in machine ethics, the advantages and challenges of each. These lecture notes are structured as follows.

A longer version of this abstract can be found at https://drops.dagstuhl.de/opus/volltexte/2022/16004/

Tutorial Overview. In this tutorial, first we discuss what is decision making and how decision-making is distinguished from moral decision-making. Decisions are made by an agent. Next we discuss what computational agents are, what does it mean for a computational agent to be autonomous and what kind of moral agents can computational agents be. One way to automate moral reasoning is to follow a specific moral theory. We give a very quick overview of what is a moral theory and some of the more known moral theories from moral philosophy. Next, In we discuss two general approaches to building artificial moral agents, we discuss open research problems and challenges.

Tutorial Scope. In this tutorial we do discussed specific examples of artificial moral agents. This tutorial is not intended to be a systematic review of implemented machine ethics systems. A very practical reason for avoiding discussing implementations of artificial agents here is that these implementations vary vastly in the approaches they use and considerable background knowledge in various reasoning and learning methods would be necessary to understand the implementations. However, references are given to these specific systems and the interested reader can follow them and explore them for learning more.

References

1. Anderson, M., Anderson, S.L.: The status of machine ethics: a report from the AAAI symposium. Minds Mach., **17**(1), 1–10 (2007). doi:10.1007/s11023-007-9053-7
2. Anderson, M., Anderson, S.L., et al.: Machine Ethics. Cambridge University Press (2011)
3. Bellman, R.E.: An Introduction to Artificial Intelligence: Can Computers Think? Boyd & Fraser Publishing Company (1978)
4. Brożek, B., Janik, B.: Can artificial intelligences be moral agents? New Ideas in Psychology, **54**, 101–106 (2019). URL: https://www.sciencedirect.com/science/article/pii/S0732118X17300739, doi:https://doi.org/10.1016/j.newideapsych.2018.12.002
5. Etzioni, A., Etzioni, O.: Incorporating ethics into artificial intelligence. J. Ethics, **21**, 403–418, (2017)
6. Fieser, J.: Ethics. In: Boylan, M., (eds.) Internet Encyclopedia of Philosophy. ISSN 2161-0002, (2021)
7. Hare, R.M.: Community and Communication, pp. 109–115. Macmillan Education UK, London, (1972). doi:10.1007/978-1-349-00955-8_9
8. Malle, B.F., Bello, P., Scheutz, M.: Requirements for an artificial agent with norm competence. In: Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society, AIES 2019, pp. 21–27, New York, NY, USA. Association for Computing Machinery (2019). doi:10.1145/3306618.3314252.
9. Wallach, W., Allen, C.: Moral machines: teaching robots right from wrong. Oxford University Press, Inc., USA (2008)

Contents