Higher-Order Game Theory

Paulo Oliva

Queen Mary University of London
p.oliva@qmul.ac.uk

Types and Topology
A workshop in honour of Martín Escardó's 60th birthday

Birmingham, 17-18 December 2025

WoLLiC 2007 - Rio de Janeiro - BR

WoLLiC 2007 - Rio de Janeiro - BR

Proof Theory

Spector's bar recursive interpretation of countable choice

$$\varepsilon\colon (A^+ \to A^-) \to A^+$$

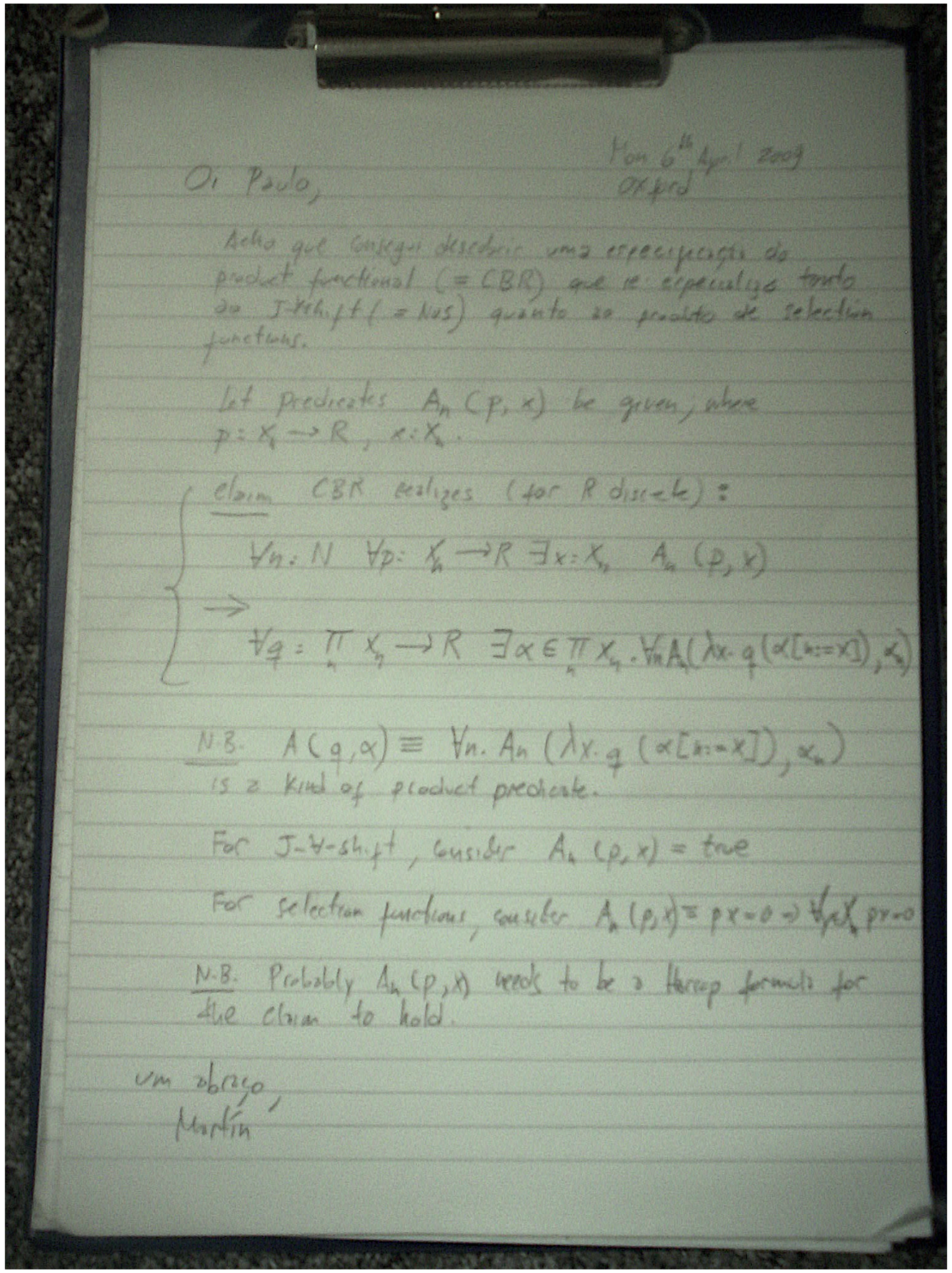
$$\overset{?}{\simeq}$$

Computability Topology

Combination of selection functions in Tychonoff's Theorem

$$\varepsilon\colon (X \to \mathbb{B}) \to X$$

eureka.pdf



Mon 6th April 2009
Oxford

Oi Paulo,

Acho que consegui descobrir uma especificação do product functional (= CBR) que se especializa tanto ao JK-shift (= KUS) quanto ao produto de selection functions.

Let predicates $A_n(p, x)$ be given, where $p : X_n \to R$, $x : X_n$.

<u>claim</u> CBR realizes (for R discrete):

$$\left\{ \begin{array}{l} \forall n : N \ \forall p : X_n \to R \ \exists x : X_n \ A_n(p, x) \\ \to \\ \forall q : \prod_n X_n \to R \ \exists \alpha \in \prod_n X_n . \forall n A_n(\lambda x . q(\alpha[n := x]), \alpha_n) \end{array} \right.$$

<u>N.B.</u> $A(q, \alpha) \equiv \forall n . A_n(\lambda x . q(\alpha[n := x]), \alpha_n)$ is a kind of product predicate.

For J-∀-shift, consider $A_n(p, x) = \mathrm{true}$

For selection functions, consider $A_n(p, x) \equiv px = 0 \Rightarrow \forall y^{X_n}(py = 0)$

<u>N.B.</u> Probably $A_n(p, x)$ needs to be a Harrop formula for the claim to hold.

um abraço,
Martín

Hi Paulo, Mon 6th April 2009
Oxford

I think I've found a specification of the product functional (CBR) which generalises the JK-shift and the product of selection functions.

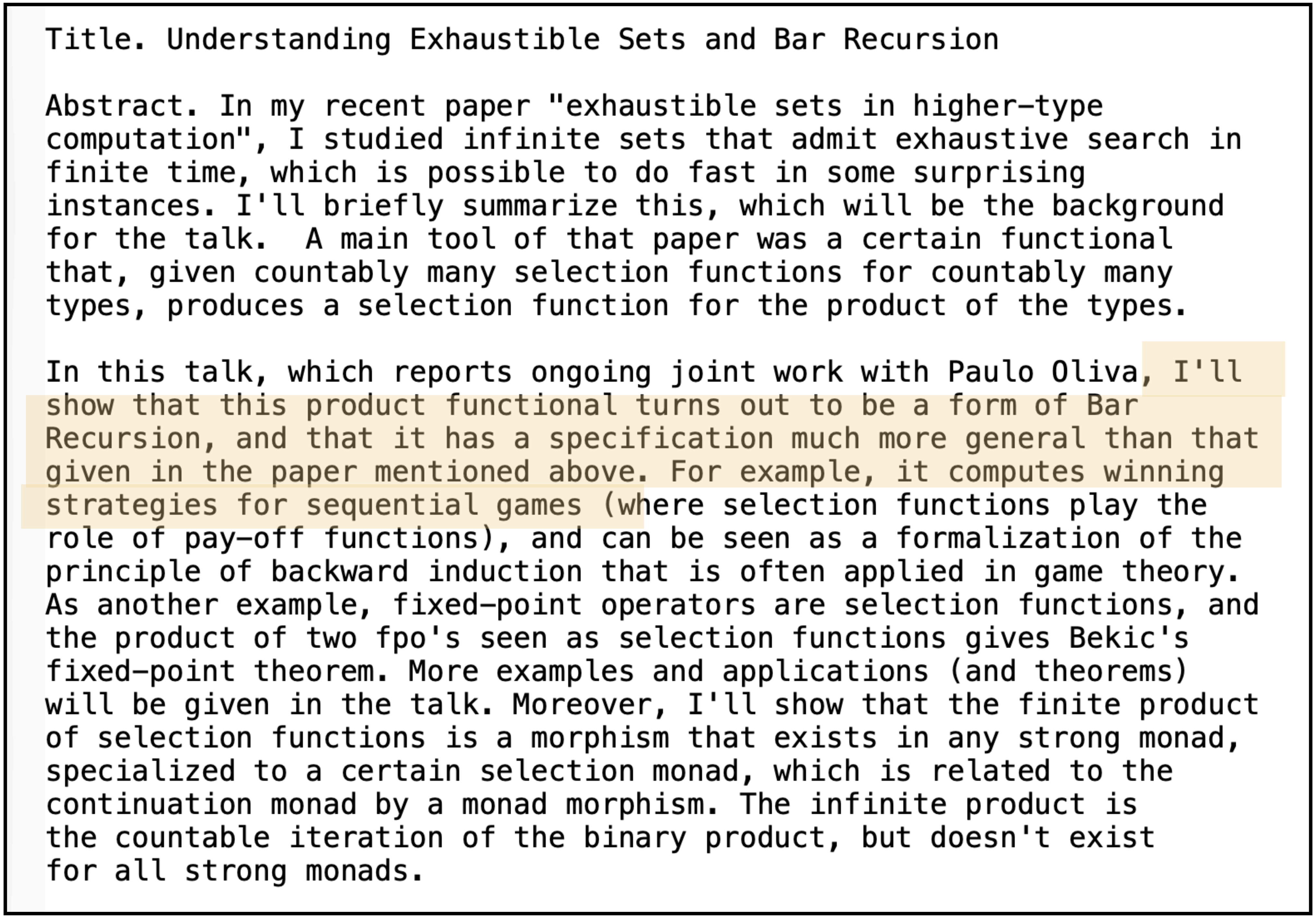
Let predicates $A_n(p, x)$ be given, where $p : X_n \to R$ and $x : X_n$

<u>Claim</u>: CBR realizes (for R discrete):

$$\left\{ \begin{array}{l} \forall n^{\mathbb{N}} \forall p^{X_n \to R} \exists x^{X_n} A_n(p, x) \\ \quad \Rightarrow \\ \quad \forall q \exists \alpha \forall n A_n(\lambda x . q(\alpha[n := x]), \alpha_n) \end{array} \right.$$

…

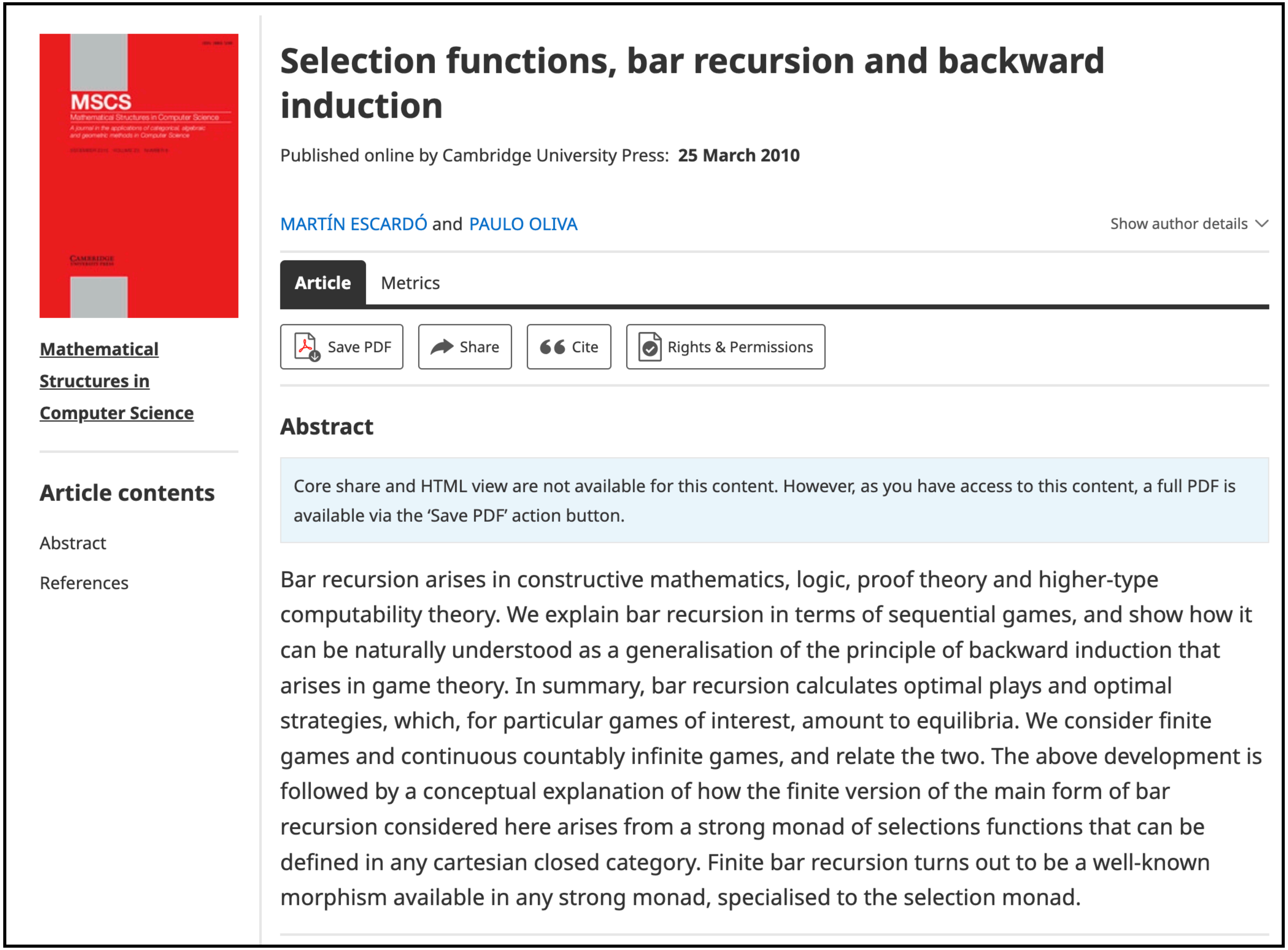
Martin’s talk @ QMUL, May 2009



Title. Understanding Exhaustible Sets and Bar Recursion

Abstract. In my recent paper "exhaustible sets in higher-type computation", I studied infinite sets that admit exhaustive search in finite time, which is possible to do fast in some surprising instances. I'll briefly summarize this, which will be the background for the talk. A main tool of that paper was a certain functional that, given countably many selection functions for countably many types, produces a selection function for the product of the types.

In this talk, which reports ongoing joint work with Paulo Oliva, I'll show that this product functional turns out to be a form of Bar Recursion, and that it has a specification much more general than that given in the paper mentioned above. For example, it computes winning strategies for sequential games (where selection functions play the role of pay-off functions), and can be seen as a formalization of the principle of backward induction that is often applied in game theory. As another example, fixed-point operators are selection functions, and the product of two fpo's seen as selection functions gives Bekic's fixed-point theorem. More examples and applications (and theorems) will be given in the talk. Moreover, I'll show that the finite product of selection functions is a morphism that exists in any strong monad, specialized to a certain selection monad, which is related to the continuation monad by a monad morphism. The infinite product is the countable iteration of the binary product, but doesn't exist for all strong monads.



Mathematical Structures in Computer Science

Selection functions, bar recursion and backward induction

Published online by Cambridge University Press: **25 March 2010**

MARTÍN ESCARDÓ and PAULO OLIVA

Article contents

Abstract

References

Abstract

Core share and HTML view are not available for this content. However, as you have access to this content, a full PDF is available via the 'Save PDF' action button.

Bar recursion arises in constructive mathematics, logic, proof theory and higher-type computability theory. We explain bar recursion in terms of sequential games, and show how it can be naturally understood as a generalisation of the principle of backward induction that arises in game theory. In summary, bar recursion calculates optimal plays and optimal strategies, which, for particular games of interest, amount to equilibria. We consider finite games and continuous countably infinite games, and relate the two. The above development is followed by a conceptual explanation of how the finite version of the main form of bar recursion considered here arises from a strong monad of selections functions that can be defined in any cartesian closed category. Finite bar recursion turns out to be a well-known morphism available in any strong monad, specialised to the selection monad.

A Tale of Two Monads...

$$\boxed{K_R X = (X \to R) \to R}$$

continuation monad

$$\eta^K_X \colon X \to K_R X$$

$$\beta^K_{X,Y} \colon K_R X \to (X \to K_R Y) \to K_R Y$$

$$\otimes^K_{X,Y} \colon K_R X \times K_R Y \to K_R(X \times Y)$$

$$\boxed{J_R X = (X \to R) \to X}$$

selection monad

$$\eta^J_X \colon X \to J_R X$$

$$\beta^J_{X,Y} \colon J_R X \to (X \to J_R Y) \to J_R Y$$

$$\otimes^J_{X,Y} \colon J_R X \times J_R Y \to J_R(X \times Y)$$

A Tale of Two Monads...

$$\phi : (X \to R) \to R$$

quantifier

$$\varepsilon : (X \to R) \to X$$

selection function

$$\phi(p) = p(\varepsilon(p))$$

attainability
(ε attains ϕ)

Higher-Order Games

Definition (Escardó/O.'2010).

A sequential higher-order game of n-rounds is a tuple $(R, (X_i)_{i<n}, q, (\phi_i)_{i<n})$ where:

- R is the type of outcomes
- X_i is the type of moves at round i
- $q \colon \Pi_{i<n} X_i \to R$ is the outcome function
- $\phi_i \colon (X_i \to R) \to R$ describe the "goal" at round i

Example 1 (SAT).

- $R = \mathbb{B}$
- $X_i = \mathbb{B}$
- $q(\vec{x}) \colon \mathbb{B}$ some Boolean formula over n atomic propositions
- $\phi_i = \exists$, existential quantifiers $((\mathbb{B} \to \mathbb{B}) \to \mathbb{B})$

Higher-Order Games

Definition (Escardó/O.'2010).

A sequential higher-order game of n-rounds is a tuple $(R, (X_i)_{i<n}, q, (\phi_i)_{i<n})$ where:

- R is the type of outcomes
- X_i is the type of moves at round i
- $q \colon \Pi_{i<n} X_i \to R$ is the outcome function
- $\phi_i \colon (X_i \to R) \to R$ describe the "goal" at round i

Example 2 (maximising n-players).

- $R = \mathbb{R}^n$
- X_i some choice of actions for player i
- $q(\vec{x}) \colon \mathbb{R}^n$ the game's payoff function (i-th player receives $\pi_i(q(\vec{x}))$)
- $\phi_i = \max_i$, maximising players $((X_i \to \mathbb{R}^n) \to \mathbb{R}^n)$

Higher-Order Games

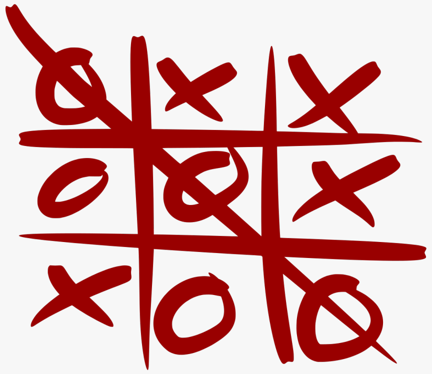
Definition (Escardó/O.'2010).

A sequential higher-order game of n-rounds is a tuple $(R, (X_i)_{i<n}, q, (\phi_i)_{i<n})$ where:

- R is the type of outcomes
- X_i is the type of moves at round i
- $q \colon \Pi_{i<n} X_i \to R$ is the outcome function
- $\phi_i \colon (X_i \to R) \to R$ describe the "goal" at round i

Example 3 (tic-tac-toe).

- $R = \{-1, 0, 1\}$
- $X_i = \{1, \ldots, 9\}$ — grid position
- $q(\vec{x})$ decides outcome of game
- $\phi_{2i} = \max$ and $\phi_{2i+1} = \min$



Higher-Order Games

Definition (Escardó/O.'2010).

A sequential higher-order game of n-rounds is a tuple $(R, (X_i)_{i<n}, q, (\phi_i)_{i<n})$ where:

- R is the type of outcomes
- X_i is the type of moves at round i
- $q \colon \Pi_{i<n} X_i \to R$ is the outcome function
- $\phi_i \colon (X_i \to R) \to R$ describe the "goal" at round i

Definition (Escardó/O.'2010).

Given a higher-order game of n-rounds is a tuple $(R, (X_i)_{i<n}, q, (\phi_i)_{i<n})$

- $(\otimes_i \phi_i)(q) \colon R$ is the optimal outcome of the game
- A strategy $\sigma_i \colon \Pi_{j<i} X_j \to X_i$ is optimal if for all $x_1, \ldots, x_{i-1}$

$$q(x_1, \ldots, x_{i-1}, \sigma) = \phi_i(\lambda x_i' \,.\, q(x_1, \ldots, x_{i-1}, x_i', \sigma))$$

Higher-Order Games

R outcomes

X_i moves

$q \colon \Pi_i X_i \to R$ outcome function

$\varphi_i \colon (X_i \to R) \to R$ goal functions

$\varepsilon_i \colon (X_i \to R) \to X_i$ selection functions

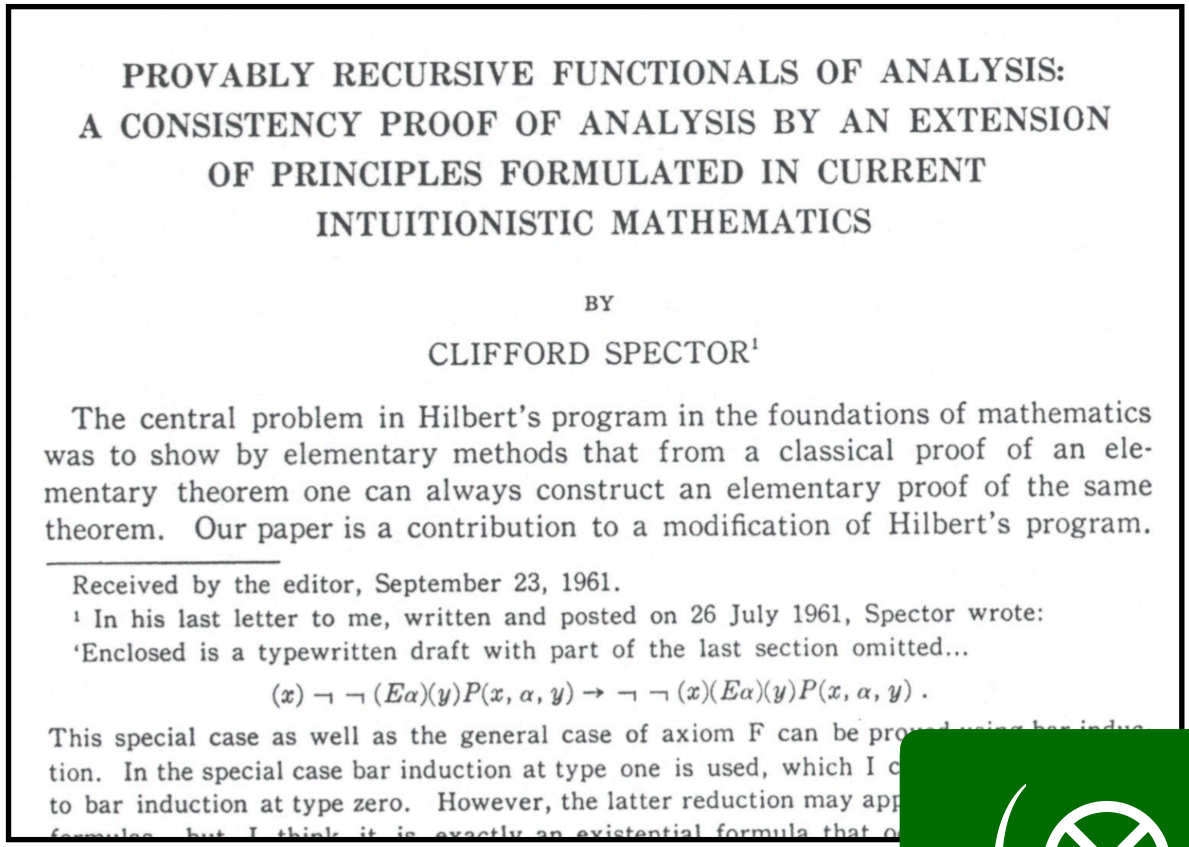
$\left(\bigotimes \varphi_i\right)(q)$

optimal outcome

$\left(\bigotimes \varepsilon_i\right)(q)$

optimal strategy

Higher-Order Games



PROVABLY RECURSIVE FUNCTIONALS OF ANALYSIS: A CONSISTENCY PROOF OF ANALYSIS BY AN EXTENSION OF PRINCIPLES FORMULATED IN CURRENT INTUITIONISTIC MATHEMATICS

BY

CLIFFORD SPECTOR[1]

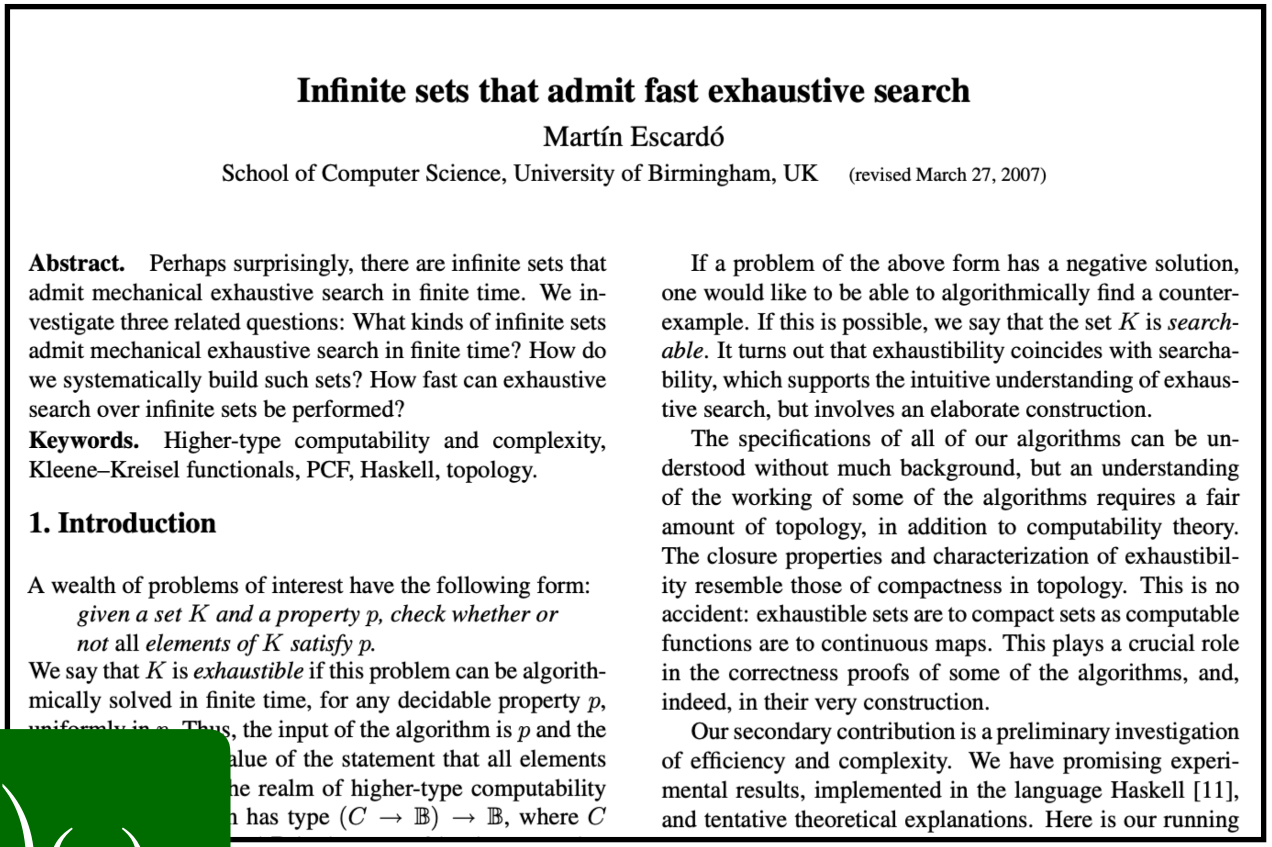
The central problem in Hilbert's program in the foundations of mathematics was to show by elementary methods that from a classical proof of an elementary theorem one can always construct an elementary proof of the same theorem. Our paper is a contribution to a modification of Hilbert's program.

Received by the editor, September 23, 1961.

[1] In his last letter to me, written and posted on 26 July 1961, Spector wrote: 'Enclosed is a typewritten draft with part of the last section omitted...

$$(x) \neg\neg (E\alpha)(y)P(x,\alpha,y) \rightarrow \neg\neg (x)(E\alpha)(y)P(x,\alpha,y) \,.$$

This special case as well as the general case of axiom F can be pro tion. In the special case bar induction at type one is used, which I c to bar induction at type zero. However, the latter reduction may app



Infinite sets that admit fast exhaustive search

Martín Escardó
School of Computer Science, University of Birmingham, UK (revised March 27, 2007)

Abstract. Perhaps surprisingly, there are infinite sets that admit mechanical exhaustive search in finite time. We investigate three related questions: What kinds of infinite sets admit mechanical exhaustive search in finite time? How do we systematically build such sets? How fast can exhaustive search over infinite sets be performed?

Keywords. Higher-type computability and complexity, Kleene–Kreisel functionals, PCF, Haskell, topology.

1. Introduction

A wealth of problems of interest have the following form:

> *given a set K and a property p, check whether or not* all *elements of K satisfy p.*

We say that K is *exhaustible* if this problem can be algorithmically solved in finite time, for any decidable property p, uniformly in p. Thus, the input of the algorithm is p and the alue of the statement that all elements he realm of higher-type computability n has type $(C \to \mathbb{B}) \to \mathbb{B}$, where C

If a problem of the above form has a negative solution, one would like to be able to algorithmically find a counterexample. If this is possible, we say that the set K is *searchable*. It turns out that exhaustibility coincides with searchability, which supports the intuitive understanding of exhaustive search, but involves an elaborate construction.

The specifications of all of our algorithms can be understood without much background, but an understanding of the working of some of the algorithms requires a fair amount of topology, in addition to computability theory. The closure properties and characterization of exhaustibility resemble those of compactness in topology. This is no accident: exhaustible sets are to compact sets as computable functions are to continuous maps. This plays a crucial role in the correctness proofs of some of the algorithms, and, indeed, in their very construction.

Our secondary contribution is a preliminary investigation of efficiency and complexity. We have promising experimental results, implemented in the language Haskell [11], and tentative theoretical explanations. Here is our running

$$\left(\bigotimes \varepsilon_i\right)(q)$$

optimal strategy

Selection Monad Transformer

$$T\colon \mathrm{Type} \to \mathrm{Type}$$

strong monad

$$\alpha\colon TR \to R$$

T-algebra

$$\varepsilon\colon (X \to R) \to TX$$

monadic selection

M. Escardó and P. Oliva
The Herbrand functional interpretation of the double negation shift
The Journal of Symbolic Logic, 82(2):590-607, 2017

M. Escardó and P. Oliva
Selection functions, bar recursion and backward induction
Mathematical Structures in Computer Science, 20(2):127-168, 2010

M. Escardó and P. Oliva
Computational interpretations of analysis via products of selection functions
Proceedings of CiE 2010

M. Escardó and P. Oliva
The Peirce translation and the double negation shift
Proceedings of CiE 2010

M. Escardó and P. Oliva
Searchable sets, Dubuc-Penon comp., omniscience princ., and drinker paradox
CiE 2010 (Abstracts and handouts booklet)

M. Escardó and P. Oliva
What sequential games, the Tychnoff theorem and the DNS have in common
MSFP 2010, ACM SIGPLAN, 21-32, 2010

M. Escardó, P. Oliva and T. Powell
System T and the product of selection functions
Computer Science Logic, 2011

M. Escardó and P. Oliva
Sequential games and optimal strategies
Proceedings of the Royal Society A, 467:1519-1545, 2011

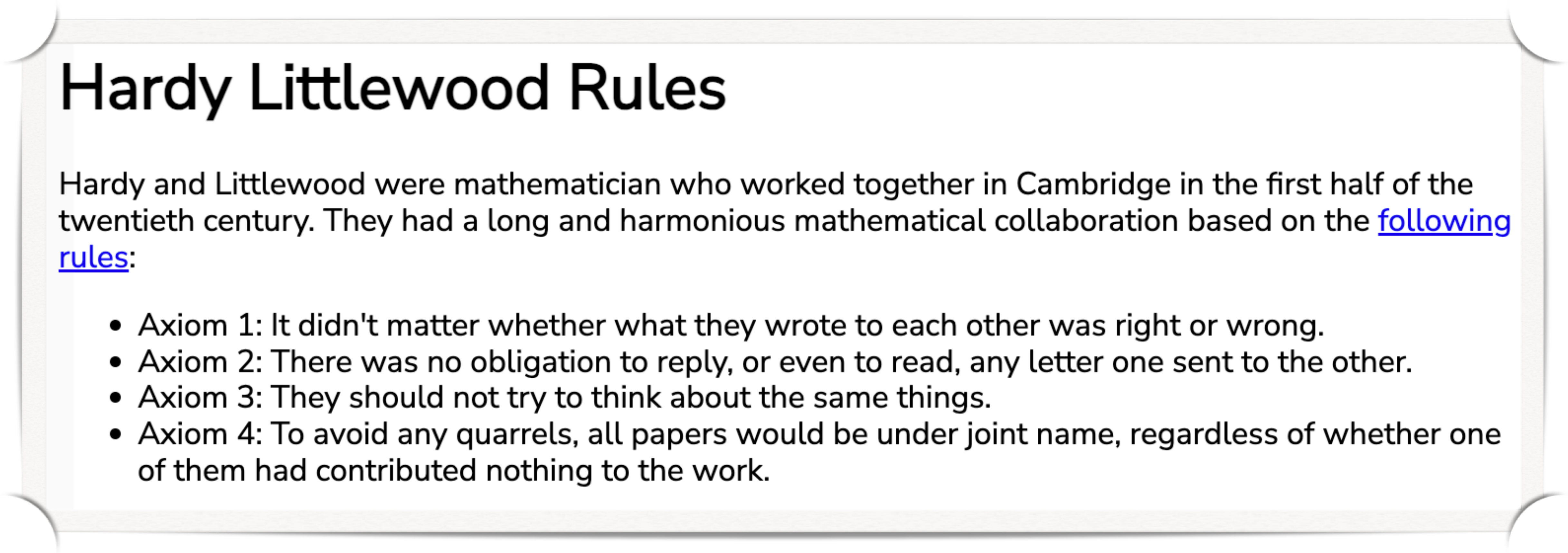
M. Escardó and P. Oliva
The Peirce translation
Annals of Pure and Applied Logic, 163(6):681-692, 2012

M. Escardó and P. Oliva
Computing Nash equilibria of unbounded games
The Turing Centenary Conference, EPiC Series, vol. 10, 53-65, Manchester, 2012

M. Escardó and P. Oliva
Bar recursion and products of selection functions
The Journal of Symbolic Logic, 80(1):1-28, 2015

M. Escardó and P. Oliva
The Herbrand functional interpretation of the double negation shift
The Journal of Symbolic Logic, 82(2):590-607, 2017

M. Escardó and P. Oliva
Higher-order games with dependent types
Theoretical Computer Science, vol 974, 2023



Hardy Littlewood Rules

Hardy and Littlewood were mathematician who worked together in Cambridge in the first half of the twentieth century. They had a long and harmonious mathematical collaboration based on the following rules:

- Axiom 1: It didn't matter whether what they wrote to each other was right or wrong.
- Axiom 2: There was no obligation to reply, or even to read, any letter one sent to the other.
- Axiom 3: They should not try to think about the same things.
- Axiom 4: To avoid any quarrels, all papers would be under joint name, regardless of whether one of them had contributed nothing to the work.

Happy Birthday, Martín!