Selection Functions and Attainable Quantifiers

Paulo Oliva

Queen Mary, University of London

(joint work with Martín Escardó)

LogIC Seminar, Imperial College

26 Nov 2009

Outline

Outline

Usual quantifiers

$$\exists, \forall \; : \; (X \to \mathbb{B}) \to \mathbb{B}$$

Usual quantifiers

$$\exists, \forall \ : \ (X \to R) \to R$$

Usual quantifiers

$$\exists, \forall : (X \to R) \to R$$

Other operations of this type are

X	R	Operation
$\mathbb{N}$	Y	Limit $\lim$
$[0,1]$	$\mathbb{R}$	Supremum $\sup$
$[0,1]$	$\mathbb{R}$	Integration $\int$
Y	Y	Fixed point operator fix_Y

Definition (Generalised Quantifiers)

Let us call operations ϕ of type

$$(X \to R) \to R$$

generalised quantifiers. Abbreviate $KX \;:\equiv\; (X \to R) \to R$.

Definition (Generalised Quantifiers)

Let us call operations ϕ of type

$$(X \to R) \to R$$

generalised quantifiers. Abbreviate $KX \; :\equiv \; (X \to R) \to R$.

Definition (Product of Generalised Quantifiers)

Given quantifiers $\phi \colon KX$ and $\psi \colon KY$ define a quantifier $\phi \otimes \psi \colon K(X \times Y)$ as

$$(\phi \otimes \psi)(p) \overset{R}{:\equiv} \phi(\lambda x^X . \psi(\lambda y^Y . p(x, y)))$$

where $p \colon X \times Y \to R$.

Generalised Quantifiers

What does

$$(\phi \otimes \psi)(p) \stackrel{R}{:\equiv} \phi(\lambda x^X.\psi(\lambda y^Y.p(x,y)))$$

mean?

Generalised Quantifiers

What does

$$(\phi \otimes \psi)(p) :\stackrel{R}{\equiv} \phi(\lambda x^X.\psi(\lambda y^Y.p(x, y)))$$

mean?

Exactly what you would expect, namely

$$(\exists_X \otimes \forall_Y)(p^{X \times Y \to \mathbb{B}}) \stackrel{\mathbb{B}}{\equiv} \exists x^X \forall y^Y p(x, y)$$

$$(\sup \otimes \textstyle\int)(p^{[0,1]^2 \to \mathbb{R}}) \stackrel{\mathbb{R}}{\equiv} \sup_x \int_0^1 p(x, y)dy$$

Outline

Theorem (Mean Value Theorem)

For any $p\colon C[0,1]$ *there is a point* $a \in [0,1]$ *such that*

$$\int_0^1 p = p(a)$$

Theorem (Mean Value Theorem)

For any $p \colon C[0,1]$ *there is a point* $a \in [0,1]$ *such that*

$$\int_0^1 p = p(a)$$

Theorem (Supremum Theorem)

For any $p \colon C[0,1]$ *there is a point* $a \in [0,1]$ *such that*

$$\sup p = p(a)$$

Theorem (Witness Theorem)

For any $p \colon X \to \mathbb{B}$ there is a point $a \in X$ such that

$$\exists x^X p(x) \quad \Leftrightarrow \quad p(a)$$

(similar to Hilbert's ε-term).

Theorem (Witness Theorem)

For any $p \colon X \to \mathbb{B}$ there is a point $a \in X$ such that

$$\exists x^X p(x) \quad \Leftrightarrow \quad p(a)$$

(similar to Hilbert's ε-term).

Theorem (Counter-example Theorem)

For any $p \colon X \to \mathbb{B}$ there is a point $a \in X$ such that

$$\forall x^X p(x) \quad \Leftrightarrow \quad p(a)$$

(aka "Drinker's paradox").

Let $JX \equiv (X \to R) \to X$.

Let $JX \equiv (X \to R) \to X$.

Definition (Selection Functions)

A function $\varepsilon \colon JX$ is called a *selection function* for $\phi \colon KX$ if

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

holds for all $p \colon X \to R$.

Let $JX \equiv (X \to R) \to X$.

Definition (Selection Functions)

A function $\varepsilon \colon JX$ is called a *selection function* for $\phi \colon KX$ if

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

holds for all $p \colon X \to R$.

Definition (Attainable Quantifiers)

A generalised quantifier $\phi \colon KX$ is called *attainable* if it has a *selection function*.

For Instance

Any fixed point operator

$$\mathsf{fix} \quad : \quad (X \to X) \to X$$

is an attainable quantifier, and a selection function.

In fact,

$$\mathsf{fix}\ p = p(\mathsf{fix}\ p)$$

says that fix is its own selection function.

A Mapping $J \mapsto K$

Not all quantifiers are attainable, but every element

$$\varepsilon \quad : \quad JX$$

is a selection function for some attainable quantifier, namely

$$\overline{\varepsilon} \quad : \quad KX$$

defined as

$$\overline{\varepsilon}p = p(\varepsilon p).$$

So, we call elements $\varepsilon \colon JX$ “selection functions”.

Questions

Is “being attainable” closed under finite product?

What about countable product?

Questions

Is "being attainable" closed under finite product?

What about countable product?

Yes! Let us define a product of selection functions such that

$$\overline{\varepsilon \otimes \delta} = \overline{\varepsilon} \otimes \overline{\delta}$$

Definition (Product of Selection Functions)

Given selection functions $\varepsilon\colon JX$ and $\delta\colon JY$ define a selection function on the product space $X \times Y$ as

$$(\varepsilon \otimes \delta)(p^{X \times Y \to R}) \stackrel{X \times Y}{:\equiv} (a, b(a))$$

where

$$\begin{array}{rcl} a & = & \varepsilon(\lambda x.p(x, b(x))) \\ b(x) & = & \delta(\lambda y.p(x, y)). \end{array}$$

For instance: Quantifier Elimination

Suppose $\exists p = p(\varepsilon p)$ and $\forall p = p(\delta p)$.

For instance: Quantifier Elimination

Suppose $\exists p = p(\varepsilon p)$ and $\forall p = p(\delta p)$. Then

$$\exists x \forall y \, p(x,y) \quad = \quad \exists x \, p(x, b(x))$$

where

$$b(x) \quad = \quad \delta(\lambda y . p(x,y))$$

For instance: Quantifier Elimination

Suppose $\exists p = p(\varepsilon p)$ and $\forall p = p(\delta p)$. Then

$$\begin{aligned} \exists x \forall y\, p(x,y) &= \exists x\, p(x, b(x)) \\ &= p(a, b(a)) \end{aligned}$$

where

$$\begin{aligned} b(x) &= \delta(\lambda y.p(x,y)) \\ a &= \varepsilon(\lambda x.p(x, b(x))). \end{aligned}$$

For instance: Quantifier Elimination

Suppose $\exists p = p(\varepsilon p)$ and $\forall p = p(\delta p)$. Then

$$\begin{array}{rcl} \exists x \forall y\, p(x,y) & = & \exists x\, p(x, b(x)) \\ & = & p(a, b(a)) \end{array}$$

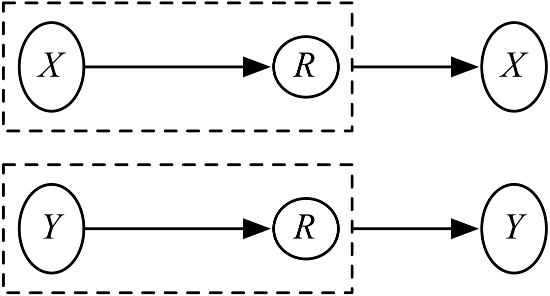
where

$$\begin{array}{rcl} b(x) & = & \delta(\lambda y.p(x,y)) \\ a & = & \varepsilon(\lambda x.p(x, b(x))). \end{array}$$

In fact, $(\varepsilon \otimes \delta)(p) = (a, b(a))$.

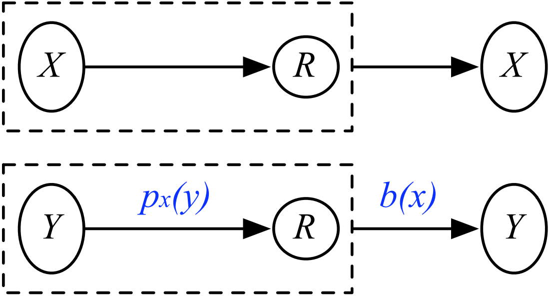
Product of Selection Functions

$$p \colon X \times Y \to R$$



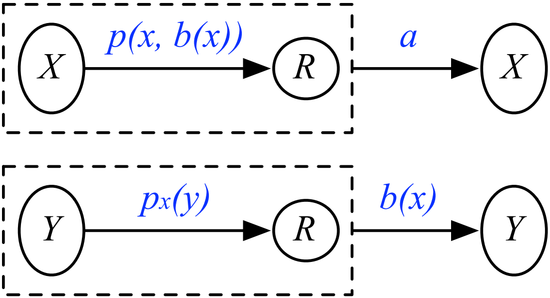
Product of Selection Functions

$$p\colon X \times Y \to R$$



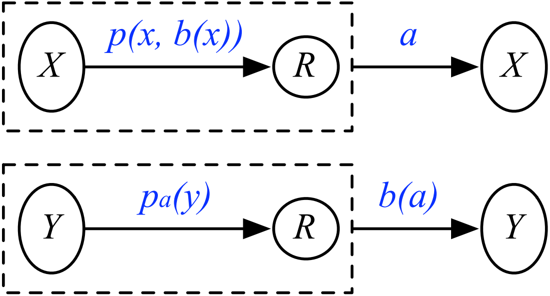
Product of Selection Functions

$$p\colon X \times Y \to R$$



Product of Selection Functions

$$p\colon X \times Y \to R$$



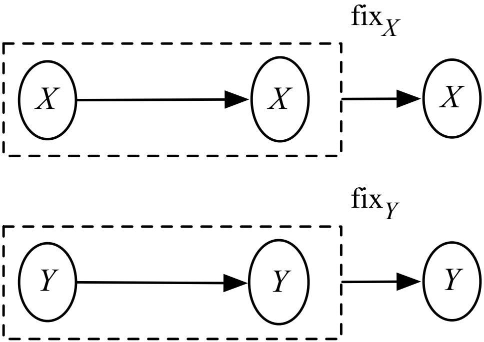
Bekic's lemma

Lemma

If X and Y have fixed point operators then so does $X \times Y$.

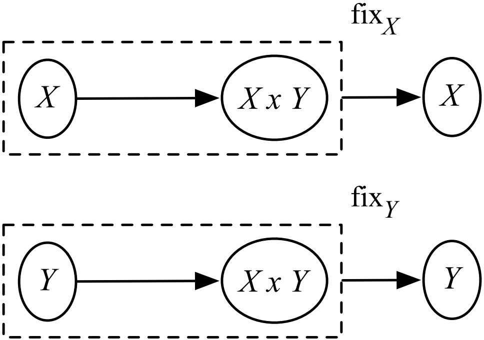
Bekic's lemma

$$p\colon X \times Y \to X \times Y$$



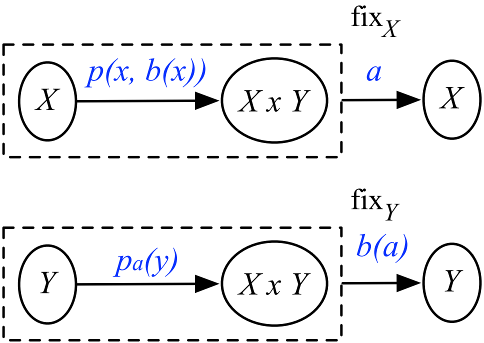
Bekic's lemma

$$p \colon X \times Y \to X \times Y$$



Bekic's lemma

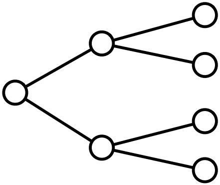
$$p\colon X \times Y \to X \times Y$$



Outline

Backtracking

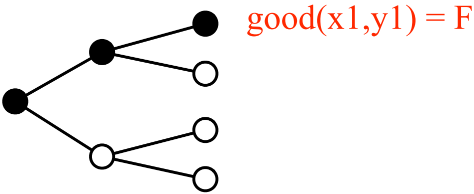
good: $X \times Y \to \mathbb{B}$



Generic algorithm has type $(X \times Y \to \mathbb{B}) \to X \times Y$.

Backtracking

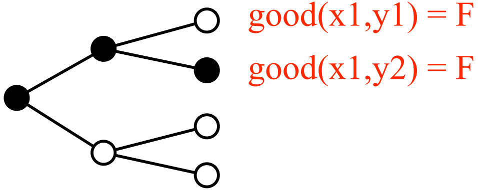
good: $X \times Y \to \mathbb{B}$



Generic algorithm has type $(X \times Y \to \mathbb{B}) \to X \times Y$.

Backtracking

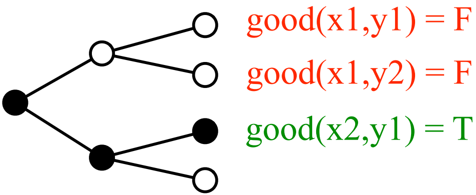
good: $X \times Y \to \mathbb{B}$



Generic algorithm has type $(X \times Y \to \mathbb{B}) \to X \times Y$.

Backtracking

$$\mathsf{good}\colon X \times Y \to \mathbb{B}$$



Generic algorithm has type $(X \times Y \to \mathbb{B}) \to X \times Y$.

Eight Queens Problem

$\varepsilon(p)$ {
  for $(i := 1; i \leq 8; i{+}{+})$ do
    if $p(i)$ return i
  return 1
}

$[\, \varepsilon \colon (8 \to \mathbb{B}) \to 8 \,]$

Eight Queens Problem

$\varepsilon(p)$ {
  for $(i := 1; i \leq 8; i{+}{+})$ do
    if $p(i)$ return i
  return 1
}

$[\ \varepsilon \colon (8 \to \mathbb{B}) \to 8\ ]$

$\mathsf{sol}_i(x_1, ..., x_{i-1})$ {
  if $i = 9$ return $\langle\ \rangle$
  else
    $y := \varepsilon(\lambda i.\mathsf{good}(\mathsf{sol}_{i+1}(x_1, \ldots, x_{i-1}, i)))$
    return $y * \mathsf{sol}_{i+1}(x_1, \ldots, x_{i-1}, y)$
}

$[\ \mathsf{sol}_i \colon 8^{i-1} \to 8^{9-i}\ ]$

Eight Queens Problem

$\varepsilon(p)$ {
  for $(i := 1; i \leq 8; i{+}{+})$ do
    if $p(i)$ return i
  return 1
}

$[\, \varepsilon \colon (8 \to \mathbb{B}) \to 8 \,]$

$\mathsf{sol}_i(x_1, ..., x_{i-1})$ {
  if $i = 9$ return $\langle\ \rangle$
  else
    $y := \varepsilon(\lambda i.\mathsf{good}(\mathsf{sol}_{i+1}(x_1, \ldots, x_{i-1}, i)))$
    return $y * \mathsf{sol}_{i+1}(x_1, \ldots, x_{i-1}, y)$
}

$[\, \mathsf{sol}_i \colon 8^{i-1} \to 8^{9-i} \,]$

$\langle x_1, \ldots, x_8 \rangle := \mathsf{sol}_1(\ )$

Eight Queens Problem

$\text{good} \colon 8^8 \to \mathbb{B}$ checks if argument is solution to 8QP.

Eight Queens Problem

good: $8^8 \to \mathbb{B}$ checks if argument is solution to 8QP.

Selection function

$$\varepsilon \colon (8 \to \mathbb{B}) \to 8$$

finds argument $\varepsilon p \in 8$ such that $p(\varepsilon p)$ holds

Eight Queens Problem

good: $8^8 \to \mathbb{B}$ checks if argument is solution to 8QP.

Selection function

$$\varepsilon \colon (8 \to \mathbb{B}) \to 8$$

finds argument $\varepsilon p \in 8$ such that $p(\varepsilon p)$ holds

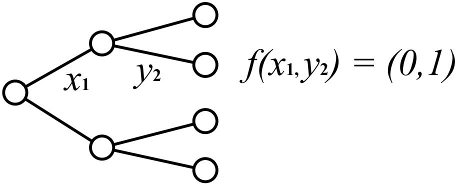
$$\mathsf{sol}_1(\,) = \left(\bigotimes_{i=1}^{8} \varepsilon \right) (\mathsf{good})$$

calculates a solution to 8 queen problem.

Outline

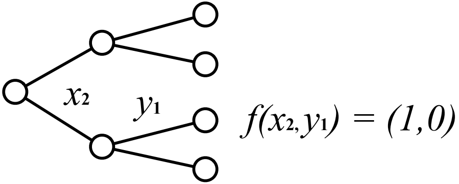
Nash equilibrium (for sequential games)

$$f : X \times Y \to \mathbb{R}^2$$



Nash equilibrium (for sequential games)

$$f : X \times Y \to \mathbb{R}^2$$



Backward Induction

Let $f\colon \Pi_{i=1}^n X_i \to \mathbb{R}^n$ be a payoff function

$\mathrm{argmax}_i(p)$ { $[\mathrm{argmax}_i \colon (X_i \to \mathbb{R}^n) \to X_i]$

  for $(x \in X_i)$ do

    if $p(x)$ has maximal i-coordinate return x

}

Backward Induction

Let $f\colon \Pi_{i=1}^{n} X_i \to \mathbb{R}^n$ be a payoff function

$\mathrm{argmax}_i(p)$ { $\qquad$ $[\mathrm{argmax}_i \colon (X_i \to \mathbb{R}^n) \to X_i]$
  for $(x \in X_i)$ do
    if $p(x)$ has maximal i-coordinate return x
}

$\mathsf{sol}_i(x_1, \ldots, x_{i-1})$ { $\qquad$ $[\mathsf{sol}_i \colon \Pi_{k=1}^{i-1} X_k \to \Pi_{k=i}^{n} X_k]$
  if $i = n + 1$ return $\langle\, \rangle$
  else
    $y := \mathrm{argmax}_i(\lambda x. f(\mathsf{sol}_{i+1}(x_1, \ldots, x_{i-1}, x)))$
    return $y * \mathsf{sol}_{i+1}(x_1, \ldots, x_{i-1}, y)$
}

Backward Induction

Let $f \colon \Pi_{i=1}^{n} X_i \to \mathbb{R}^n$ be a payoff function

$\mathrm{argmax}_i(p)$ { $\qquad$ $[\mathrm{argmax}_i \colon (X_i \to \mathbb{R}^n) \to X_i]$
for $(x \in X_i)$ do
if $p(x)$ has maximal i-coordinate return x
}

$\mathsf{sol}_i(x_1, \ldots, x_{i-1})$ { $\qquad$ $[\mathsf{sol}_i \colon \Pi_{k=1}^{i-1} X_k \to \Pi_{k=i}^{n} X_k]$
if $i = n+1$ return $\langle\,\rangle$
else
$y := \mathrm{argmax}_i(\lambda x. f(\mathsf{sol}_{i+1}(x_1, \ldots, x_{i-1}, x)))$
return $y * \mathsf{sol}_{i+1}(x_1, \ldots, x_{i-1}, y)$
}

$\langle x_1, \ldots, x_n \rangle := \mathsf{sol}_1(\,)$

Backward Induction

Payoff function $f\colon \Pi_{i=1}^{n} X_i \to \mathbb{R}^n$

Each selection function

$$\mathrm{argmax}_i\colon (X_i \to \mathbb{R}^n) \to X_i$$

finds a point where the argument is i-maximal

Product

$$\mathsf{sol}_1(\ ) = \left(\bigotimes_{i=1}^{n} \mathrm{argmax}_i \right)(f)$$

calculates a strategy profile in Nash equilibrium.

Outline

Analysis

Mathematical analysis is based on **comprehension**

$$\exists f \forall n (fn = 0 \leftrightarrow A(n)).$$

Analysis

Mathematical analysis is based on **comprehension**

$$\exists f \forall n (fn = 0 \leftrightarrow A(n)).$$

Comprehension follows classically from **countable choice**

$$\forall n \exists b A(n, b) \rightarrow \exists f \forall n A(n, fn).$$

Analysis

Mathematical analysis is based on **comprehension**

$$\exists f \forall n (fn = 0 \leftrightarrow A(n)).$$

Comprehension follows classically from **countable choice**

$$\forall n \exists b A(n, b) \to \exists f \forall n A(n, fn).$$

Countable choice is classically computational up to $\mathbf{DNS}$

$$\forall n \neg\neg A(n) \to \neg\neg \forall n A(n).$$

Double negation shift

The double negation shift **DNS**

$$\forall n \neg\neg A(n) \rightarrow \neg\neg \forall n A(n)$$

corresponds to the type

$$\Pi_n((A_n \rightarrow \perp) \rightarrow \perp) \rightarrow (\Pi_n A_n \rightarrow \perp) \rightarrow \perp .$$

Double negation shift

The double negation shift $\mathbf{DNS}$

$$\forall n \neg\neg A(n) \to \neg\neg \forall n A(n)$$

corresponds to the type

$$\Pi_n((A_n \to \perp) \to \perp) \to (\Pi_n A_n \to \perp) \to \perp .$$

If $\perp \to A_n$, this is equivalent to

$$\Pi_n((A_n \to \perp) \to A_n) \to (\Pi_n A_n \to \perp) \to \Pi_n A_n$$

i.e. $\Pi_n J(A_n) \to J(\Pi_n A_n)$.

Double negation shift

The double negation shift $\mathbf{DNS}$

$$\forall n \neg\neg A(n) \to \neg\neg \forall n A(n)$$

corresponds to the type

$$\Pi_n((A_n \to \perp) \to \perp) \to (\Pi_n A_n \to \perp) \to \perp .$$

If $\perp \to A_n$, this is equivalent to

$$\Pi_n((A_n \to \perp) \to A_n) \to (\Pi_n A_n \to \perp) \to \Pi_n A_n$$

i.e. $\Pi_n J(A_n) \to J(\Pi_n A_n)$.

The type of the **countable product** of selection functions!

Bar recursion

Bar recursion does precisely that, i.e. it can be viewed as

$$\bigotimes_{n} \varepsilon = \varepsilon_n \otimes \bigotimes_{n+1} (\varepsilon).$$

Bar recursion

Bar recursion does precisely that, i.e. it can be viewed as

$$\bigotimes_{n} \varepsilon = \varepsilon_n \otimes \bigotimes_{n+1}(\varepsilon).$$

Spector's bar recursive solution to consistency of analysis is

$$\bigotimes_{s}(\varepsilon) = \begin{cases} \mathbf{0} & \text{if } \omega_s(\mathbf{0}) < |s| \\ \varepsilon_s \otimes \lambda x.(\bigotimes_{s*x}(\varepsilon)) & \text{otherwise.} \end{cases}$$

Not Mentioned but Very Interesting

- Connection to **classical logic**
  Finite product of quantifiers witnesses dialectica interpretation of IPHP
- General notion of **game**
  Optimal strategies as products of selection functions
  History dependent games, dependent products
- Relation to **monads**
  K, J are strong monads, $\varepsilon \mapsto \overline{\varepsilon}$ a monad morphism

Not Mentioned but Very Interesting

- Connection to **classical logic**
  Finite product of quantifiers witnesses dialectica interpretation of IPHP
- General notion of **game**
  Optimal strategies as products of selection functions
  History dependent games, dependent products
- Relation to **monads**
  K, J are strong monads, $\varepsilon \mapsto \overline{\varepsilon}$ a monad morphism

For more information see:

Selection functions, bar recursion and backward induction
M. Escardo and P. Oliva, MSCS, to appear.