Selection Functions, Bar Recursion and Nash Equilibrium

Paulo Oliva

Queen Mary, University of London, UK

(based on joint work with Martín Escardó)

British Logic Colloquium

Swansea, 4 September 2009

Outline

Outline

Usual quantifiers

$$\exists_X, \forall_X \; : \; (X \to \mathbb{B}) \to \mathbb{B}$$

Usual quantifiers ($R = \mathbb{B}$)

$$\exists_X, \forall_X \ : \ (X \to R) \to R$$

Usual quantifiers ($R = \mathbb{B}$)

$$\exists_X, \forall_X \ : \ (X \to R) \to R$$

Some operations of this type:

Operation	ϕ	:	$(X \to R) \to R$
Quantifiers	$\forall_X, \exists_X$	:	$(X \to \mathbb{B}) \to \mathbb{B}$
Integration	$\int_0^1$	:	$([0,1] \to \mathbb{R}) \to \mathbb{R}$
Supremum	$\sup_{[0,1]}$	:	$([0,1] \to \mathbb{R}) \to \mathbb{R}$
Limit	$\lim$	:	$(\mathbb{N} \to R) \to R$
Fixed point operator	fix_X	:	$(X \to X) \to X$

Definition (Generalised Quantifiers)

Let us call operations ϕ of type

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

generalised quantifiers. Write $K_R X \; :\equiv \; (X \to R) \to R$.

Definition (Generalised Quantifiers)

Let us call operations ϕ of type

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

generalised quantifiers. Write $K_R X \; :\equiv \; (X \to R) \to R$.

Definition (Product of Generalised Quantifiers)

Given quantifiers $\phi \colon K_R X$ and $\psi \colon K_R Y$ define the **product quantifier** $\phi \otimes \psi \colon K_R(X \times Y)$ 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

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

mean?

Exactly what you would expect, namely

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

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

Outline

Theorem (Mean Value Theorem)

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

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

Theorem (Mean Value Theorem)

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

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

Theorem (Maximum Theorem)

For any $p \in C[0,1]$ *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 $J_R X :\equiv (X \to R) \to X$.

Let $J_R X :\equiv (X \to R) \to X$.

Definition (Selection Functions)

$\varepsilon\colon J_R X$ is called a **selection function** for $\phi\colon K_R X$ if

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

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

Let $J_R X :\equiv (X \to R) \to X$.

Definition (Selection Functions)

$\varepsilon\colon J_R X$ is called a **selection function** for $\phi\colon K_R X$ if

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

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

Definition (Attainable Quantifiers)

A generalised quantifier $\phi\colon K_R X$ is called **attainable** if it has a selection function $\varepsilon\colon J_R X$.

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, the fixed point equation

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

says that fix is its own selection function.

A Mapping $J_R \mapsto K_R$

Not all quantifiers are attainable, but every element

$$\varepsilon \quad : \quad J_R X$$

is a selection function for some attainable quantifier, namely

$$\overline{\varepsilon} \quad : \quad K_R X$$

defined as

$$\overline{\varepsilon} p :\stackrel{R}{=} p(\varepsilon p).$$

So, we call all 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! We 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 $\varepsilon \colon J_R X$ and $\delta \colon J_R Y$ define a **product selection function**

$$\varepsilon \otimes \delta \quad : \quad J_R(X \times Y)$$

as

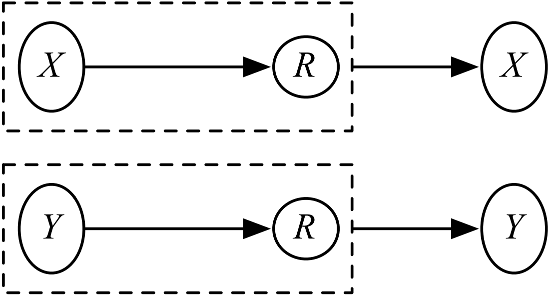
$$(\varepsilon \otimes \delta)(p^{X \times Y \to R}) \overset{X \times Y}{:=} (a, b(a))$$

where

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

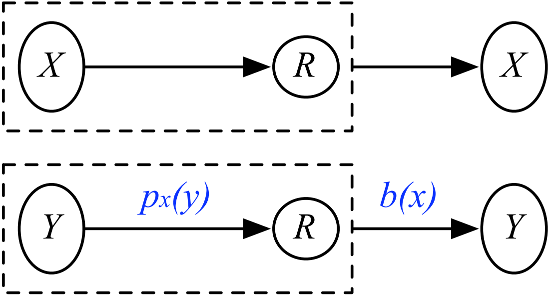
Product of Selection Functions

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



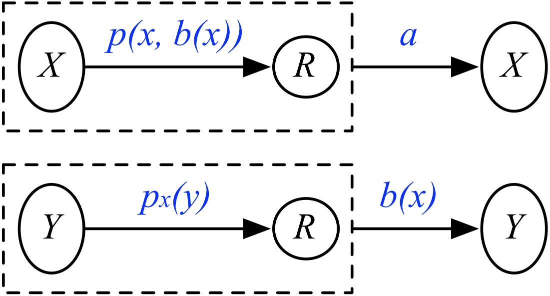
Product of Selection Functions

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



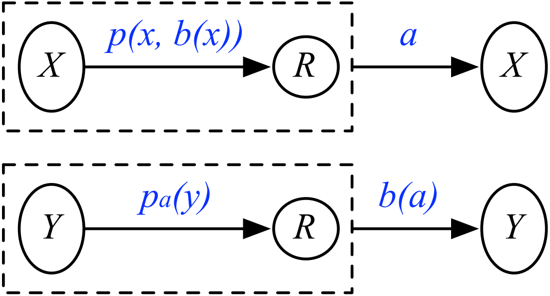
Product of Selection Functions

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



Product of Selection Functions

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



Quantifier Elimination

Suppose $\exists n\, p(\vec{v}, n) = p(\vec{v}, \varepsilon(\lambda n.p(\vec{v}, n))$.

Quantifier Elimination

Suppose $\exists n\, p(\vec{v}, n) = p(\vec{v}, \varepsilon(\lambda n.p(\vec{v}, n)))$. Then

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

where

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

Quantifier Elimination

Suppose $\exists n\, p(\vec{v}, n) = p(\vec{v}, \varepsilon(\lambda n.p(\vec{v}, n))$. Then

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

where

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

Quantifier Elimination

Suppose $\exists n\, p(\vec{v}, n) = p(\vec{v}, \varepsilon(\lambda n. p(\vec{v}, n))$. Then

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

where

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

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

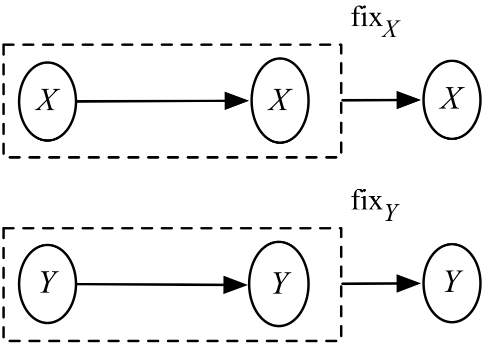
Bekic's lemma

Lemma

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

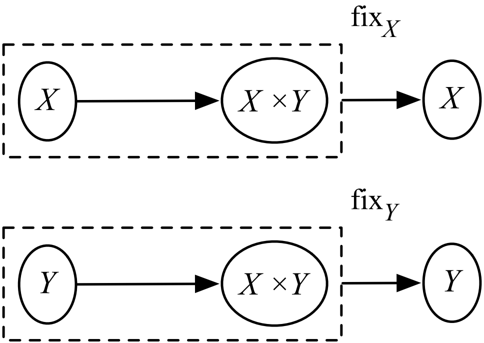
Bekic's lemma

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



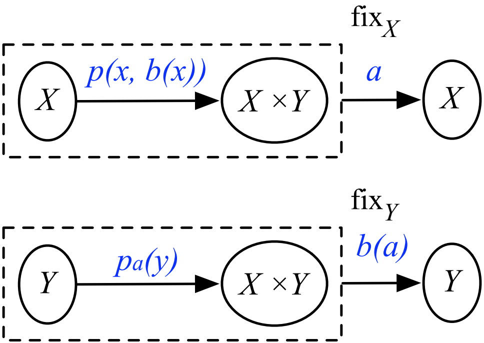
Bekic's lemma

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



Bekic's lemma

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



Outline

Nash equilibrium (simultaneous games)

- n players, each with a set of "strategies" X_i
- **payoff function** $f\colon \Pi_{i=0}^{n-1} X_i \to \mathbb{R}^n$
- **strategy profile** $(x_0, \ldots, x_{n-1})\colon \Pi_{i=0}^{n-1} X_i$

Nash equilibrium (simultaneous games)

- n players, each with a set of "strategies" X_i
- **payoff function** $f\colon \Pi_{i=0}^{n-1} X_i \to \mathbb{R}^n$
- **strategy profile** $(x_0, \ldots, x_{n-1})\colon \Pi_{i=0}^{n-1} X_i$
- **equilibrium strategy profile** if for $i = 0, \ldots, n-1$
  $\forall x_i^* (f_i(x_0, \ldots, x_i^*, \ldots, x_{n-1}) \leq f_i(x_0, \ldots, x_i, \ldots, x_{n-1}))$

Nash equilibrium (simultaneous games)

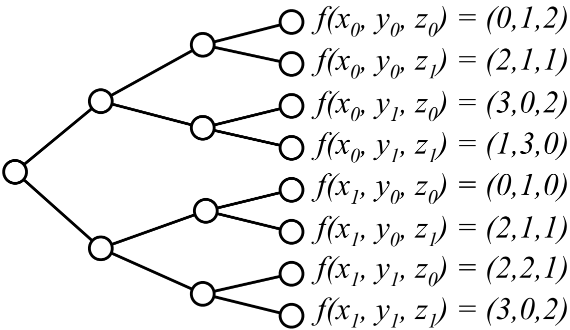
- n players, each with a set of "strategies" X_i
- **payoff function** $f \colon \Pi_{i=0}^{n-1} X_i \to \mathbb{R}^n$
- **strategy profile** $(x_0, \ldots, x_{n-1}) \colon \Pi_{i=0}^{n-1} X_i$

- **equilibrium strategy profile** if for $i = 0, \ldots, n-1$
  $\forall x_i^* (f_i(x_0, \ldots, x_i^*, \ldots, x_{n-1}) \leq f_i(x_0, \ldots, x_i, \ldots, x_{n-1}))$

- pure equilibria not always exist, but mixed ones do
- consider, however, that the game is played **sequentially**

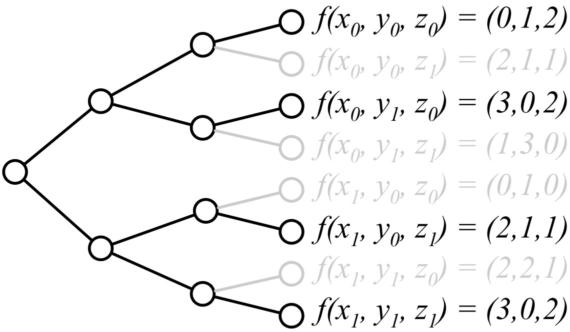
Nash equilibrium (for sequential games)

E.g. three players, payoff function $f\colon X \times Y \times Z \to \mathbb{R}^3$



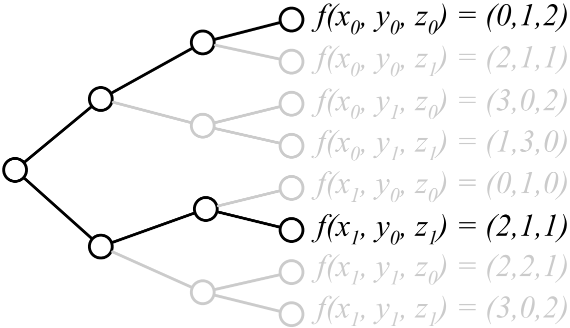
Nash equilibrium (for sequential games)

E.g. three players, payoff function $f\colon X \times Y \times Z \to \mathbb{R}^3$



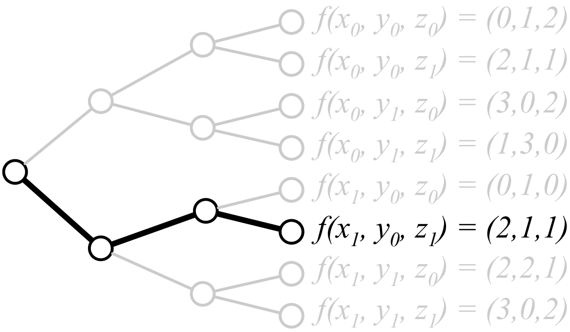
Nash equilibrium (for sequential games)

E.g. three players, payoff function $f\colon X \times Y \times Z \to \mathbb{R}^3$



Nash equilibrium (for sequential games)

E.g. three players, payoff function $f\colon X \times Y \times Z \to \mathbb{R}^3$



Backward Induction

Selection functions in this case are

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

for $(x \in X_i)$ do

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

}

Backward Induction

Selection functions in this case are

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

for $(x \in X_i)$ do

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

}

Product

$$\left(\bigotimes_{i=0}^{n-1} \mathrm{argmax}_i\right)(f)$$

computes "optimal play", and can be used to calculate strategy profile in Nash equilibrium.

Outline

Bar recursion = infinite product

Bar recursion is simply the countable iteration of product of selection functions and quantifiers!

Bar recursion = infinite product

Bar recursion is simply the countable iteration of product of selection functions and quantifiers!

In other words, define **infinite product** as

$$\bigotimes_{k}(\varepsilon) = \varepsilon_k \otimes \left(\bigotimes_{k+1}(\varepsilon)\right).$$

where $\varepsilon \colon \Pi_{k \in \mathbb{N}} J_R(X_k)$.

Bar recursion = infinite product

Bar recursion is simply the countable iteration of product of selection functions and quantifiers!

In other words, define **infinite product** as

$$\bigotimes_{k}(\varepsilon) = \varepsilon_k \otimes \Big(\bigotimes_{k+1}(\varepsilon)\Big).$$

where $\varepsilon \colon \Pi_{k \in \mathbb{N}} J_R(X_k)$.

Then (*intuitively*)

$$\mathsf{BR}(\varepsilon, p, s) = \bigotimes_{|s|}(\varepsilon)(p_s).$$

Two points

Point 1. Infinite products not always (uniquely) defined.

Recursive equation uniquely defines a function in the model of *continuous functionals*.

But it does not on the *full set theoretic model*.

Two points

Point 1. Infinite products not always (uniquely) defined.

Recursive equation uniquely defines a function in the model of *continuous functionals*.

But it does not on the *full set theoretic model*.

Point 2. There are several variants of bar recursion, but only two binary products have been defined?

Product of quant.	$\mapsto$	Spector BR [Spector'62]
Product of s.f.	$\mapsto$	Course-of-value BR [Escardo/O.'09]
Skewed product	$\mapsto$	Modified BR [Berger/O.'06]
Symmetric product	$\mapsto$	BBC [Berardi et al'98]

Spector Bar recursion

Iterated product of quantifiers

$$\bigotimes_{k}(\phi) = \phi_k \otimes \left(\bigotimes_{k+1}(\phi)\right)$$

in general fails to exist (even assuming continuity).

Spector Bar recursion

Iterated product of quantifiers

$$\bigotimes_k(\phi) = \phi_k \otimes \left(\bigotimes_{k+1}(\phi)\right)$$

in general fails to exist (even assuming continuity).

Spector's original bar recursion corresponds to a "conditional" iterated product

$$\bigotimes_k(\phi)(p) \stackrel{\mathbb{N}}{=} \begin{cases} p(\mathbf{0}) & \text{if } p(\mathbf{0}) < k \\ (\phi_k \otimes (\bigotimes_{k+1}(\phi)))(p) & \text{otherwise.} \end{cases}$$

Ps.: Actually, Spector uses **dependent products** – c.f. paper.

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

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

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 **DNS**

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

corresponds to the type

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

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

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

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

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

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
- **Interdefinability** between bar recursions
  E.g. "normal" product = "skewed" product

For more information see:

Selection functions, bar recursion and backward induction

M. Escardo and P. Oliva, Submitted, July 2009

Preprint available from my webpage.

Instances of bar recursion as iterated products of selection functions and quantifiers

M. Escardo and P. Oliva, In preparation.