# Realizzabilità classica ed ultrafiltri su $\mathbb{N}$

Davide Barbarossa

Université d'Aix-Marseille/Università di Roma Tre

*davide94barbarossa@gmail.com*

20/07/2018

# Corrispondenza di Curry-Howard

| Logica | Linguaggio di programmazione | Esecuzione |
|---|---|---|
| Intuizionista | $\lambda$-calcolo/ dimostrazioni intuizioniste | $\beta$-riduzione/ cut-elimination |
| Classica | $\lambda_c$-calcolo + istruzioni/ dimostrazioni classiche + assiomi | Krivine Abstract Machine |

$$\text{Formule/tipi} ::= X(e) \mid A \to B \mid \forall x A \mid \forall X A$$

# Il tipaggio

Quasi-prove del $\lambda_c$-calcolo: $\mathrm{QP} ::= \; x \mid \mathsf{callcc} \mid (t)u \mid \lambda x t$

Regole di tipaggio:

$$\frac{}{\Gamma, x : A \vdash x : A} \qquad \frac{}{\Gamma \vdash \mathsf{callcc} : ((A \to B) \to A) \to A}$$

$$\frac{\Gamma \vdash t : A \to B \qquad \Gamma \vdash u : A}{\Gamma \vdash (t)u : B} \qquad \frac{\Gamma, x : A \vdash t : B}{\Gamma \vdash \lambda x t : A \to B}$$

$$\frac{\Gamma \vdash t : A}{\Gamma \vdash t : \forall x A} \; (x \textit{ non} \text{ libera in } A) \qquad \frac{\Gamma \vdash t : A}{\Gamma \vdash t : \forall X A} \; (X \textit{ non} \text{ libera in } A)$$

$$\frac{\Gamma \vdash t : \forall x A}{\Gamma \vdash t : A[e/x]} \qquad \frac{\Gamma \vdash t : \forall X^k A}{\Gamma \vdash t : A[B/X x_1, \ldots, x_k]}$$

# Il linguaggio di programmazione e l'esecuzione

## $\lambda_c$-calcolo

(Programmi) $\Lambda_c ::= \; x \mid \text{callcc} \mid \kappa_\pi \mid (t)u \mid \lambda xt$
(Pile) $\Pi_c ::= \; \alpha \mid t.\pi$ con $t, u \in \Lambda_c, \; \pi \in \Pi_c$
(Processi) $\Lambda \star \Pi := \Lambda_c \times \Pi_c$

## KAM per il $\lambda_c$-calcolo

(push) $(t)u \star \pi \succ t \star u.\pi;$

(grab) $\lambda xt \star u.\pi \succ t[u/x] \star \pi;$

(save) $\text{callcc} \star t.\pi \succ t \star \kappa_\pi.\pi;$

(restore) $\kappa_\pi \star t.\rho \succ t \star \pi.$

# Un esempio

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{x : \neg\neg A, y : \neg A \vdash x : \neg A \to \bot} \qquad \dfrac{}{x : \neg\neg A, y : \neg A \vdash y : \neg A}}{x : \neg\neg A, y : \neg A \vdash (x)y : \bot}}{x : \neg\neg A, y : \neg A \vdash (x)y : A}}{x : \neg\neg A \vdash \lambda y(x)y : \neg A \to A} \qquad \dfrac{}{x : \neg\neg A \vdash \mathsf{callcc} : (\neg A \to A) \to A}}{\dfrac{x : \neg\neg A \vdash (\mathsf{callcc})\lambda y(x)y : A}{\vdash \lambda x(\mathsf{callcc})\lambda y(x)y : \neg\neg A \to A}}
$$

$$
\lambda x(\mathsf{callcc})\lambda y(x)y \star \xi . \pi \succ \mathsf{callcc} \star \lambda y(\xi)y . \pi \succ \lambda y(\xi)y \star \kappa_\pi . \pi \succ \xi \star \kappa_\pi . \pi
$$

# Interpretazioni di realizzabilità

"Realizzare" un assioma $A =$ trovare un ($\lambda_c$-)programma $t$ che giustifica A

## Semantica di realizzabilità

Si fissano:

- un segmento iniziale $\perp\!\!\!\perp \subseteq (\Lambda_c \times \Pi_c, \succ)$
- un'interpretazione in $\mathbb{N}$ delle espressioni $e$ al primo ordine ed un'interpretazione in $\mathscr{P}(\Pi_c)^{\mathbb{N}^k}$ delle variabili al secondo ordine

Si definiscono $\|.\| : \mathcal{F}_{par} \to \mathscr{P}(\Pi_c)$ e $|.| : \mathcal{F}_{par} \to \mathscr{P}(\Lambda_c)$ come segue:

$$\|\mathcal{H}[e]\| := \mathcal{H}(e)$$

$$\|A \to B\| := |A|.\|B\|$$

$$\|\forall x A\| := \bigcup_{a \in \mathbb{N}} \|A[a/x]\|$$

$$\|\forall X A\| := \bigcup_{\mathcal{H} \in \mathscr{P}(\Pi_c)^{\mathbb{N}^k}} \|A[\mathcal{H}/X]\|$$

$$|A| := \{\xi \in \Lambda_c \text{ t.c. } \forall \pi \in \|A\|,\ \xi \star \pi \in \perp\!\!\!\perp\}$$

Si pone $t \Vdash A$ come abbreviazione di $t \in |A|$.

# Esempi ed osservazioni

1. $\perp\!\!\!\perp = \Lambda_c \times \Pi_c \;\Rightarrow\; \exists t \in \mathrm{QP} \subseteq \Lambda_c$ t.c. $t \Vdash \bot$
2. $\forall \pi \in \|A\|,\ \kappa_\pi \Vdash \neg A$
3. Sappiamo che: $\vdash \lambda x(\mathsf{callcc})\lambda y(x)y : \neg\neg A \to A$.
   Si ha anche: $\lambda x(\mathsf{callcc})\lambda y(x)y \Vdash \neg\neg A \to A$, infatti:

$$\begin{cases} \lambda x(\mathsf{callcc})\lambda y(x)y \star \xi . \pi \succ \xi \star \kappa_\pi . \pi \\ \begin{cases} \pi \in \|A\| \\ \xi \Vdash \neg A \to \bot \end{cases} \quad \Rightarrow \quad \xi \star \kappa_\pi . \pi \in \perp\!\!\!\perp \end{cases}$$

# Alcuni risultati di base

**Teorema (Lemma di adequazione)**

$$\vdash t : A \;\Rightarrow\; t \Vdash A$$

**Proposizione**

$$\forall t \in \mathrm{QP},\; t \nVdash \forall x Nat\{x\}$$

**Semantica di realizzabilità $\rightsquigarrow$ Modelli alla Tarski**

La teoria $\{A \text{ formula t.c. } \exists t \in \mathrm{QP} \subseteq \Lambda \text{ t.c. } t \Vdash A\}$ è coerente (sotto un'ipotesi su $\perp\!\!\!\perp$).
Quindi ammette modelli alla Tarski, chiamati modelli di realizzabilità .

# Questioni di base

### Realizzare assiomi

Data una formula $A$, trovare un programma $t$ t.c. $t \Vdash A$

Esempio: $A = \text{l'assioma dell'ultrafiltro}$

### Problema della specificazione

Dato $\vdash t : A$ allora $t \Vdash A$; descrivere *come* $t$ giustifica $A$
(Nota bene: non *cosa* fa $t$, quello è gia dato dal tipaggio $t : A$)

### Teorema (specificazione per l'identità polimorfa)

$\vdash \theta : \forall X(X \to X) \;\Rightarrow\; \theta \approx \lambda xx$ *(nel senso che* $\theta \star t.\pi \succ t \star \pi \prec \lambda xx \star t.\pi$*)*

### Teorema (specificazione per i booleani)

$\vdash \theta : Bool[1] \;\Rightarrow\; \theta \approx \lambda x\lambda yx$ *(nel senso che* $\theta \star t.u.\pi \succ t \star \pi \prec \lambda x\lambda yx \star t.u.\pi$*)*

# L'obiettivo

### Assioma dell'ultrafiltro (AU)

Esiste un ultrafiltro non triviale su $\mathbb{N}$

### Definizione

*Un insieme* $\mathcal{U} \subseteq \mathscr{P}(\mathbb{N})$ *si dice ultrafiltro non triviale su* $\mathbb{N}$ *sse* $\forall A, B \subseteq \mathbb{N}$,

$$\varnothing \notin \mathcal{U}$$

$$\mathcal{U} \ni A \subseteq B \Rightarrow B \in \mathcal{U}$$

$$A, B \in \mathcal{U} \Rightarrow A \cap B \in \mathcal{U}$$

$$A \in \mathcal{U} \Leftrightarrow \mathbb{N} \setminus A \notin \mathcal{U}.$$

### Teorema

$$\mathrm{AU} \vdash A \Rightarrow \exists \theta \text{ programma t.q. } \theta \Vdash A$$

# Astrazione della nozione di programma

### Algebra di realizzabilità

Il dato di:

- tre insiemi $\Lambda$ (programmi), $\Pi$ (pile), $\Lambda \star \Pi$ (processi)
- tre applicazioni $\mathrm{cons} : \Lambda \times \Pi \to \Pi$, $\mathrm{cont} : \Pi \to \Lambda$, $\mathrm{proc} : \Lambda \times \Pi \to \Lambda \star \Pi$
- "un modo per vedere le quasi-prove come programmi: $\mathrm{QP} \subseteq \Lambda_c \to \Lambda$"
- un *polo* $\perp\!\!\!\perp \subseteq \Lambda \star \Pi$ saturo rispetto ad una $\mathrm{KAM}$

### L'algebra $\mathrm{SR}_0$

$\Lambda := \Lambda_c$, $\Pi := \Pi_c$, $\Lambda \star \Pi := \Lambda_c \times \Pi_c$,
$\mathrm{cons}(t, \pi) := t.\pi$, $\mathrm{cont}(\pi) := \kappa_\pi$, $\mathrm{proc}(t, \pi) := (t, \pi)$,
con $\perp\!\!\!\perp$ saturo rispetto alla $\mathrm{KAM}$ per il $\lambda_c$-calcolo

# Interpretazioni di realizzabilità

1. stessa definizione di $\|A\| \subseteq \Pi$ e $|A| \subseteq \Lambda$
(e $\xi \Vdash A$ abbreviazione di $\xi \in |A|$)
2. stesso lemma di adequazione
3. si ottengono modelli di realizzabilità considerando la stessa teoria coerente

# Il forcing

### In teoria degli insiemi (P. Cohen, 1960)

- Insieme parzialmente ordinato di "condizioni di forcing" ed un filtro $G$ di "buone" condizioni
- Relazione $p \Vdash A$, per $p$ condizione
- Nuovi modelli $\mathfrak{M}[G]$ t.c. $\mathfrak{M}[G] \models A$ sse $\exists p \in G$ t.q. $p \Vdash A$
- La scelta appropriata delle condizioni permette di forzare delle formule fissate (negazione dell'ipotesi del continuo,...)

### In realizzabilità classica

**Due linguaggi formali**: $\mathscr{F}_0$ (che permette di parlare della relazione di forcing) et $\mathscr{F}_1$ (che non lo permette)

**Struttura di forcing**: $P$ preordine di condizioni di forcing,
$\mathscr{C}$ insieme di condizioni non triviali

**Trasformazione di forcing**: $P \times \mathscr{F}_1 \longrightarrow \mathscr{F}_0$, $(p, A) \to p \Vdash A$

# Giocare con il forcing

## Internalizzare il forcing

- Si definisce l'algebra $\mathrm{SR}_1$ moralmente con "$\mathrm{SR}_1 := P \times \mathrm{SR}_0$" ($\Lambda_1 := \Lambda_c \times P$, $\Pi_1 := \Pi_c \times P$, ma $\perp\!\!\!\perp_1$ più complicato...)
- Si definisce una interpretazione di realizzabilità per $\mathscr{F}_0$ in $\mathrm{SR}_0$ e per $\mathscr{F}_1$ in $\mathrm{SR}_1$

## Proposizione

$$\xi \Vdash_0 \mathscr{C}(p) \to A \;\Rightarrow\; (\tilde{\xi}, p) \Vdash_1 A$$

$$(\xi, p) \Vdash_1 A \;\Rightarrow\; \xi' \Vdash_0 \mathscr{C}(p) \to A$$

## Osservazione

*Alla base c'è una trasformazione di programmi* $(.)^* : \Lambda_c \to \Lambda_c$ *t.q.* $t^*$ *si esegue come* $t$ *ma in modalità protetta*

# Una struttura di forcing particolare

### La condizione della catena numerabile

- Una struttura di forcing soddisfa la CCD quando "tutte" le successioni "decrescenti" di condizioni non triviali ammettono un minorante non triviale
- Le strutture di forcing la soddisfano hanno delle proprietà forti (molto lavoro tecnico...)

### Proposizione

*Si può definire una struttura di forcing con insieme di condizioni l'insieme $\mathscr{P}(\Pi_c)^{\mathbb{N}}$ e che soddisfa la CCD*

### Teorema

$$\exists(\theta, p) \in \Lambda_1 \text{ t.q. } (\theta, p) \Vdash_1 \mathrm{AU}$$

# Le dimostrazioni in analisi forniscono dei programmi!

### Teorema

*Sia* $B = \text{l'axiome de choix dépendant}$. *Allora:*

$$\mathrm{AU}, B \vdash A \;\Rightarrow\; \exists \theta \in \mathrm{QP} \subseteq \Lambda_c \text{ t.q. } \theta \Vdash_0 A$$

### Idea della dimostrazione.

$\mathrm{AU} \vdash B \to A \Rightarrow (\eta, 1) \Vdash_1 B \to A \Rightarrow \eta' \Vdash_0 \mathscr{C}(1) \to B \to A \Rightarrow (\eta')\xi\nu \Vdash_0 A$, per un $\xi \Vdash_0 \mathscr{C}(1)$ (esiste sempre) e $\nu \Vdash_0 B$ (Krivine). $\square$

### Osservazione

- *Un ultrafiltro* $\mathcal{U} \subseteq \mathscr{P}(\mathbb{N})$ *su* $\mathbb{N}$ *è detto selettivo sse per tutte le partizioni* $\{\mathcal{P}_i\}_{i \in I}$ *di* $\mathbb{N}$ *t.q.* $\forall i \in I$, $\mathcal{P}_i \notin \mathcal{U}$, *si ha:* $\exists U \in \mathcal{U}$ *t.q.* $\forall i \in I$, $U \cap P_i$ *è un singleton.*
- *Possiamo anche supporre che* $\mathrm{AU}$ *dica che esiste un ultrafiltro selettivo.*

# C'est la fin!