ROMA TRE
UNIVERSITÀ DEGLI STUDI

Reti di prova in una variante della Logica Intuizionista

Candidato:
Gianluca
Calcagni

Relatore:
Dr. Lorenzo
Tortora de Falco

2 maggio 2011

Sommario

In questo lavoro di tesi introdurremo alcuni degli ultimi sviluppi della logica matematica moderna: dopo aver formalizzato il concetto di *linguaggio matematico* e di *calcolo logico*, daremo più versioni del noto teorema di *eliminazione del taglio* (il cosiddetto *Hauptsatz*) e vedremo brevemente qualche sua applicazione. Successivamente introdurremo una variante della logica lineare intuizionista chiamata *LLJ*: tale sistema sarà ancora suscettibile di eliminazione dei tagli perfino se munito di una regola apposita per la riduzione all'assurdo. Definiremo su *LLJ* la teoria delle *reti di prova* (in inglese, *proof-nets*) e discuteremo delle proprietà peculiari di queste strutture, oltre a fornire un algoritmo di ricostruzione delle *scatole* ed a dare una nuova dimostrazione dell'*Hauptsatz* **puramente geometrica**.

Dopo aver definito il calcolo delle classi ed aver esaminato alcuni dei paradossi che si possono (o *non* si possono) presentare, definiremo la teoria del λ-calcolo e sfrutteremo il *paradosso di Curry* per tipare derivazioni tramite generici λ-termini.

Capitolo 1: Logica Predicativa

Sintesi. *In questo capitolo si descrive cosa sia un* linguaggio matematico *e le proprietà che deve soddisfare. Si daranno inoltre definizioni di concetti di base quali i* predicati*, le* formule*, le* teorie *e i* modelli *di un linguaggio. Infine, si darà la definizione di linguaggio con relazione di* identità *tra termini.*

Si è soliti in matematica dare un linguaggio (del primo ordine) tramite una *segnatura*:

$$\mathcal{L} = (\mathcal{V}, \{\vee, \wedge, \exists, \forall, \mathbb{T}, \mathbb{F}, (,)\}, \mathcal{C}, \mathcal{P}, \bigcup_{n\geq 1} \mathcal{R}_n, \bigcup_{n\geq 1} \mathcal{F}_n)$$

che contenga (nell'ordine): i simboli di variabile del primo ordine x, y ecc. in $\mathcal{V}$; i simboli logici di base quali la disgiunzione $\vee$, la congiunzione $\wedge$, il quantificatore esistenziale $\exists$, la costante di "verità" $\mathbb{T}$ e così via; i simboli di costanti individuali in $\mathcal{C}$ (ad esempio, il simbolo zero per l'aritmetica di Peano); i simboli di costante proposizionale in $\mathcal{P}$ (cioé simboli di proposizioni che hanno un valore di verità fissato a priori); i simboli di predicato n-ario $\mathcal{R}_n$; i simboli di funzione n-aria $\mathcal{F}_n$. A volte, per chiarire ulteriormente di cosa stiamo parlando, aggiungeremo un simbolo per insiemi X e ridefiniremo la segnatura in modo che, ad esempio, un predicato n-ario appaia come un sottoinsieme di X^n. Indicheremo il linguaggio così ulteriormente specificato con il simbolo $\mathcal{L}_V$ (questa è l'impostazione seguita in [AF09] e nella nostra tesi).

- I *termini* sono definiti come composizione tra simboli di funzione, simboli di costante e variabili (ad esempio, se f è un simbolo binario di funzione del nostro linguaggio, allora $f(0, x)$ è un termine ben definito). I termini non hanno un valore di verità proprio, però possono essere inseriti all'interno di una proposizione (ad esempio, se $P(y)$ è la proposizione $y = 0$ allora $P(f(0, x))$ è la proposizione $f(0, x) = 0$).

- Le *formule* sono definite come composizione di constanti proposizionali, predicati applicati su termini e simboli logici: se $\mathcal{F}$ è l'insieme delle formule, allora i seguenti oggetti sintattici sono suoi elementi: $A(x, y) \wedge B(x, y, z)$ oppure $\forall z : P(f(z, z))$ ecc. Diremo che la *lunghezza* di una formula è il numero di passi necessari per definirla a partire da formule atomiche.

  In genere si definisce sull'insieme delle formule una relazione di equivalenza $\sim$ che identifichi, ad esempio, due oggetti di questo tipo:

  $$\forall x : P(x) \sim \forall y : P(y)$$

  facendo in modo che la variabile quantificata non risulti importante per la comprensione della formula. Definiremo anche il concetto di

formula *chiusa* (dove ogni occorrenza di variabile è *quantificata*, come in $\forall x : \exists y : B(x, x, y)$) e di formula *aperta* (come in $\exists x : A(x, y)$).

- Nella logica predicativa, la *negazione* di una formula si definisce direttamente sul linguaggio: stabiliremo che per ogni formula esiste la sua negazione e la definiremo in modo tale che vengano rispettate le classiche leggi di *De Morgan*, le relazioni duali sui quantificatori ed il fatto che due negazioni affermano:

$$(A \wedge B)^{\perp} = A^{\perp} \vee B^{\perp} \qquad (\exists x : A(x))^{\perp} = \forall x : A(x)^{\perp} \qquad (A^{\perp})^{\perp} = A$$

- Un *modello* di un linguaggio è, in generale, un oggetto semantico (quale, ad esempio, un preciso gruppo algebrico, un insieme o un ordinale) che "interpreta" ogni oggetto sintattico di un linguaggio $\mathcal{L}_V$ dato: si dice che il modello *interpreta* una data formula sse si può associare (in maniera univoca) a tale formula una proprietà sul modello che risulta verificata. Ad esempio, la formula $\forall x : \forall y : x + y = y + x$ viene interpretata correttamente da un modello quale $(\mathbb{Z}_2, +)$ o $(\mathbb{R}, +)$.

- Una *teoria* di un determinato linguaggio è un insieme (anche infinito) di formule chiuse costruite in tale linguaggio. Esempi tipici di teorie sono la teoria aritmetica di Peano o la teoria degli anelli. Esistono anche esempi di teorie al secondo ordine, quale può essere la teoria dei numeri reali con la continuità *à la* Dedekind.

- Un linguaggio con relazione di *identità* è esattamente quello che sembra: un linguaggio con l'aggiunta un simbolo di predicato binario Id (o '=') adibito ad identificare i termini del linguaggio.

Capitolo 2: Calcolo dei Sequenti e Logica Intuizionista

Sintesi. *In questo capitolo si introducono due importanti calcoli logici: il* calcolo dei sequenti *(o* LK*) e la* logica intuizionista *(o* LJ*). Si definisce cosa sia una* derivazione logica *e indicheremo le proprietà di base che ci aspetteremo da un calcolo logico, quali la sua* correttezza *e la sua* consistenza sintattica. *Infine, enunceremo il* teorema di completezza *per* LK.

La *deduzione naturale* [Pra65] ed il *calcolo dei sequenti* [Gen34] sono due esempi di calcolo logico introdotti da G. Gentzen nella prima metà degli anni *'30:* il secondo, in particolare, è stato il punto di riferimento dei logici matematici per tutto il ventesimo secolo. Piuttosto che definire cosa sia in generale un "calcolo logico", preferiamo descrivere direttamente il calcolo dei sequenti e ricollegarci ad esso ogni volta che ne introdurremo uno nuovo.

- Un *multiinsieme* su un insieme X dato è definito come la classe di equivalenza di una sequenza di elementi di X modulo permutazioni qualsiasi. Un *sequente* Γ su un linguaggio $\mathcal{L}_V$ è definito come un multiinsieme

finito delle sue formule. Una *presentazione*[1] $\vdash \Gamma$ di un sequente è una sequenza precisa del sequente Γ.

A differenza degli insiemi, in un multiinsieme uno stesso elemento può presentarsi più e più volte: ciò dà luogo al concetto di *occorrenza* e *molteplicità* di un elemento in un multiinsieme (ad esempio, se $\vdash \Gamma = = (\mathbb{T}, \mathbb{F}, \mathbb{T})$ allora abbiamo due distinte occorrenze del simbolo $\mathbb{T}$ ed una sola del simbolo $\mathbb{F}$; quindi $\mathbb{T}$ e $\mathbb{F}$ hanno, rispettivamente, molteplicità due ed uno).

- Definiremo le *regole di inferenza* (o, semplicemente, *regole*) del calcolo dei sequenti come la lista di regole mostrata in *Tabella 1:*

Regole Identità: *Assioma* e *Taglio*

$$\frac{\emptyset}{\vdash \underline{A}, \underline{A^\perp}}\ AX \qquad \frac{\vdash \Gamma, \overline{A} \qquad \vdash \overline{A^\perp}, \Delta}{\vdash \Gamma, \Delta}\ CUT$$

Regole Moltiplicative: $\wedge_m$ e $\vee_m$

$$\frac{\vdash \Gamma, \overline{A_1} \qquad \vdash \Delta, \overline{A_2}}{\vdash \Gamma, \Delta, \underline{A_1 \wedge A_2}}\ \wedge_m \qquad \frac{\vdash \Sigma, \overline{A_1}, \overline{A_2}}{\vdash \Sigma, \underline{A_1 \vee A_2}}\ \vee_m$$

Regole Additive: $\wedge_a$, $\vee_a^1$ e $\vee_a^2$

$$\frac{\vdash \Gamma, \overline{A_1} \qquad \vdash \Gamma, \overline{A_2}}{\vdash \Gamma, \underline{A_1 \wedge A_2}}\ \wedge_a \qquad \frac{\vdash \Delta, \overline{A_i}}{\vdash \Delta, \underline{A_1 \vee A_2}}\ \vee_a^i$$

Regole sui Quantificatori: $\exists$ e $\forall$

$$\frac{\vdash \Gamma, \overline{A(\mathbf{t}/x)}}{\vdash \Gamma, \underline{\exists x : A(x)}}\ \exists \qquad \frac{\vdash \Delta, \overline{A(x)}}{\vdash \Delta, \underline{\forall x : A(x)}}\ \forall$$

x non libera in Γ

Regole sulle unità: *Vero* e *Falso*

$$\frac{\emptyset}{\vdash \underline{\mathbb{T}}}\ \mathbb{T} \qquad \frac{\vdash \Gamma}{\vdash \Gamma, \underline{\mathbb{F}}}\ \mathbb{F}$$

Regole Strutturali: *Indebolimento* e *Contrazione*

$$\frac{\vdash \Gamma}{\vdash \Gamma, \underline{A}}\ W \qquad \frac{\vdash \Gamma, \overline{A}, \overline{A}}{\vdash \Gamma, \underline{A}}\ C$$

dove A è una formula a piacere

Tabella 1: Regole del Calcolo dei Sequenti *LK*

Ogni regola va letta dall'alto verso il basso, dove il sequente in alto (o la coppia di sequenti in alto, nel caso delle regole CUT, $\wedge_m$ e $\wedge_a$) sono le *ipotesi* ed il sequente in basso è la *conclusione* della regola (il

[1] è molto comune usare il termine 'sequente' anche per indicare una sua presentazione, sebbene la cosa sia logicamente scorretta

simbolo $\emptyset$ sta a rappresentare il sequente vuoto). Le regole che hanno il sequente vuoto come ipotesi sono dette *regole iniziali*; le regole che hanno una ipotesi non-vuota sono dette *regole unarie* e le regole che hanno due ipotesi non-vuote sono dette *regole binarie*.

Infine, diremo che che una regola *introduce* una occorrenza di formula della sua conclusione se essa appare sottolineata nella tabella data: in tal caso diremo che la formula introdotta è una *formula principale* della regola. Viceversa, le formule sopralineate sono dette *formule introduttive* o anche *formule attive* della regola (in seguito eviteremo di usare formule sotto/sopralineate per non appesantire la lettura).

- Una *derivazione logica* (o, semplicemente, *derivazione*) Γ su una teoria $\mathcal{E}$ di un linguaggio dato verrà indicata con $\mathcal{E} \vdash \Gamma$ ed è definita per induzione nel sequente modo:

  1. se Γ è la conclusione di una regola iniziale, allora $\mathcal{E} \vdash \Gamma$ per qualunque teoria $\mathcal{E}$
  2. se $A \in \mathcal{E}$, allora $\mathcal{E} \vdash A$
  3. se Γ è la conclusione di una regola unaria di ipotesi Δ e vale che $\mathcal{E} \vdash \Delta$, allora $\mathcal{E} \vdash \Gamma$
  4. se Γ è la conclusione di una regola binaria di ipotesi Δ e Σ e valgono sia $\mathcal{E} \vdash \Delta$ che $\mathcal{E} \vdash \Sigma$, allora $\mathcal{E} \vdash \Gamma$

  Una derivazione viene solitamente indicata con la lettera greca π; se π' compare nella definizione induttiva di π, diremo che π' è una *sottoderivazione* di π.

  Una derivazione logica π sulla teoria vuota viene a volta chiamata una *deduzione* del calcolo dei sequenti. Se una deduzione π dimostra il sequente $\vdash \Gamma$, diremo che Γ è il suo *risultato*.

  Esempio. *Le seguenti sono due deduzioni distinte di uguale risultato:*

$$\dfrac{\dfrac{\dfrac{\dfrac{\emptyset}{\vdash A, A^\perp}\,AX}{\vdash A, A^\perp \vee B^\perp}\,\vee_a^1 \qquad \dfrac{\dfrac{\emptyset}{\vdash B, B^\perp}\,AX}{\vdash B, A^\perp \vee B^\perp}\,\vee_a^2}{\vdash A \wedge B, A^\perp \vee B^\perp}\,\wedge_a \qquad \dfrac{\dfrac{\dfrac{\emptyset}{\vdash A, A^\perp}\,AX \qquad \dfrac{\emptyset}{\vdash B, B^\perp}\,AX}{\vdash A \wedge B, A^\perp, B^\perp}\,\wedge_m}{\vdash A \wedge B, A^\perp \vee B^\perp}\,\vee_m \tag{1}$$

- Così come abbiamo definito la relazione di equivalenza $\sim$ sulle formule del linguaggio, allo stesso modo definiremo una relazione di equivalenza $\sim_2$ sui sequenti in modo che valga la seguente:

$$\left(\vdash A_1, \ldots, A_n\right) \sim_2 \left(\vdash B_1, \ldots, B_n\right) \quad \text{sse:} \quad \forall i = 1, \ldots, n : A_i \sim B_i$$

Infine definiremo una relazione di equivalenza $\sim_3$ sulle derivazioni in modo che, per ogni passaggio induttivo della definizione, valga la equivalenza modulo $\sim_2$ tra i sequenti. Tali relazioni di equivalenza risultano utili per il seguente motivo: a volte sarà necessario operare una sostituzione sulla derivazione π per ottenere una nuova derivazione $\pi[\mathbf{t}/x]$ in cui ogni occorrenza libera della variabile x è stata sostituita con il termine $\mathbf{t}$; poiché una semplice sostituzione sintattica porterebbe a risultati scorretti, si è soliti usare prima la relazione $\sim_3$ per cambiare il nome alle variabili x vincolate (evitando così che vengano fraintese per variabili libere).

- Spesso useremo le seguenti proprietà per studiare la potenza e la espressività di un calcolo logico. Per risparmiare tempo, definiremo tali proprietà per il calcolo dei sequenti e le richiameremo in seguito ogni volta che cambieremo calcolo logico.

  1. **Reversibilità di una regola:** una regola è detta *reversibile* sse la sua/le sue ipotesi sono logicamente derivabili dalla conclusione.
  2. **Completezza di sequenti iniziali atomici:** un calcolo rispetta tale proprietà sse ogni derivazione può essere trasformata in una nuova derivazione in cui ogni regola iniziale introduce esclusivamente formule atomiche.
  3. **Commutazione di regole:** una regola $\odot$ è *discendente* su $\star$ (o, equivalentemente, $\star$ è *ascendente* su $\odot$) sse ogni derivazione in cui queste due occorrenze di regole appaiono consecutivamente può essere trasformata in una derivazione in cui le stesse due occorrenze di regole appaiono consecutivamente ma in *ordine inverso*. Una regola, in generale, sarà detta essere *discendente* (rispettivamente, *ascendente*) sse è discendente (risp. ascendente) su ogni altra regola del calcolo logico.
  4. **Proprietà della sottoformula:** una regola $\odot$ rispetta tale proprietà sse ogni formula della/delle ipotesi è anche sottoformula di qualche formula della conclusione; una derivazione π rispetta tale proprietà sse ogni sua regola la rispetta.
  5. **Consistenza sintattica:** un calcolo logico è *sintatticamente consistente* sse non esiste alcuna deduzione che dia una antinomia come risultato (ciò equivale ad affermare che $\vdash \mathbb{F}$ non è mai derivabile).

- La *logica intuizionista* (o LJ) è un calcolo logico molto simile al calcolo dei sequenti; la sua segnatura è la seguente:

$$\mathcal{L}_V = (\mathcal{V}, \{\vee, \wedge, \to, \neg, \exists, \forall, \mathbb{T}, \mathbb{F}, (,)\}, \mathcal{C}, \mathcal{P}, \bigcup_{n\geq 1} \mathcal{R}_n, \bigcup_{n\geq 1} \mathcal{F}_n)$$

le sue regole sono mostrate nella *tabella 2*.

Identità	$\dfrac{\emptyset}{A \vdash A}\ AX$		$\dfrac{\Gamma \vdash A \qquad \Delta, A \vdash B}{\Gamma, \Delta \vdash B}\ CUT$
Congiunzione	$\dfrac{\Gamma \vdash A_1 \qquad \Gamma \vdash A_2}{\Gamma \vdash A_1 \wedge A_2}\ \wedge_F$		$\dfrac{\Delta, A_i \vdash B}{\Delta, A_1 \wedge A_2 \vdash B}\ \wedge^i_R$
Disgiunzione	$\dfrac{\Gamma, A_1 \vdash B \qquad \Gamma, A_2 \vdash B}{\Gamma, A_1 \vee A_2 \vdash B}\ \vee_F$		$\dfrac{\Delta \vdash A_i}{\Delta \vdash A_1 \vee A_2}\ \vee^i_R$
Implicazione	$\dfrac{\Gamma, A \vdash B}{\Gamma \vdash A \to B}\ \to_F$		$\dfrac{\Delta \vdash A \qquad \Sigma, B \vdash X}{\Delta, \Sigma, A \to B \vdash X}\ \to_R$
Negazione	$\dfrac{\Gamma, A \vdash \mathbb{F}}{\Gamma \vdash \neg A}\ \neg_F$		$\dfrac{\Delta \vdash A}{\Delta, \neg A \vdash B}\ \neg_R$ dove B è una formula a piacere
Quantificatori	$\dfrac{\Gamma \vdash A(x)}{\Gamma \vdash \forall x : A(x)}\ \forall_F$ x non libera in Γ		$\dfrac{\Delta, A(\mathbf{t}/x) \vdash B}{\Delta, \forall x : A(x) \vdash B}\ \forall_R$
	$\dfrac{\Gamma, A(x) \vdash B}{\Gamma, \exists x : A(x) \vdash B}\ \exists_F$ x non libera in $\Gamma \cup \{B\}$		$\dfrac{\Delta \vdash A(\mathbf{t}/x)}{\Delta \vdash \exists x : A(x)}\ \exists_R$
Strutturali	$\dfrac{\Gamma, A, A \vdash B}{\Gamma, A \vdash B}\ C$		$\dfrac{\Gamma \vdash A}{\Gamma, X \vdash A}\ W$
Unità	$\dfrac{\Gamma \vdash A}{\Gamma, \mathbb{T} \vdash A}\ \mathbb{T}_F$	$\dfrac{\emptyset}{\vdash \mathbb{T}}\ \mathbb{T}_R$	$\dfrac{\emptyset}{\mathbb{F} \vdash A}\ \mathbb{F}_R$ *Ex falso, quodlibet*

Tabella 2: Regole della Logica Intuizionista LJ

Si noti come la negazione '$\neg$' non sia più costruita all'interno del linguaggio, bensì venga introdotta da una coppia di regole apposite: questo fatto ha vaste conseguenze, tra cui la mancanza delle leggi di *De Morgan* ed il fatto che una doppia negazione non equivale necessariamente ad una affermazione.

Capitolo 3: La Teoria della Normalizzazione

Sintesi. *In questo capitolo si descrive uno dei più famosi metodi per provare la consistenza sintattica di un calcolo logico: il processo di* eliminazione del taglio *(o* Hauptsatz*).*

Il teorema di eliminazione del taglio è uno dei risultati più profondi della logica matematica del secolo scorso: il suo utilizzo permette di dimostrare,

ad esempio, se un determinato calcolo logico sia sintatticamente consistente oppure se una data ipotesi del calcolo sia indipendente dal resto! Un altro utilizzo, molto più recente, è stato scovato all'interno della teoria della computazione: la corrispondenza di Curry-Howard consente infatti di associare un passo di riduzione del taglio (in un frammento della logica intuizionista) ad un passo di β-riduzione del λ-calcolo: ergo, l'esistenza (e l'unicità) di una *forma normale* di una derivazione diventa, nella corrispondenza, una dimostrazione della terminazione di un algoritmo scritto in λ-calcolo.

Teorema (Eliminazione del taglio, o *Hauptsatz*)**.** *Sia π una derivazione logica in* LK *di conclusione Γ in un dato linguaggio $\mathcal{L}_V$. É dunque possibile trasformare π in una nuova derivazione π' (avente la stessa conclusione) in cui non appaiono occorrenze della regola* CUT.

Osservazione. Esiste una versione dello stesso teorema anche in logica intuizionista.

Si noti che una derivazione priva di occorrenze del taglio rispetta automaticamente la proprietà della sottoformula: ne consegue che una dimostrazione del sequente vuoto è impossibile.

Corollario. Dal teorema di eliminazione del taglio in LK ed LJ consegue la consistenza sintattica dei due calcoli logici.

La dimostrazione del teorema di eliminazione del taglio non è banale, ma si basa su alcuni presupposti semplici:

1. è possibile definire il *rango* ed il *peso* di una occorrenza del taglio: essi sono, rispettivamente, il numero di "tagli significativi" e "contrazioni significative" da cui l'occorrenza proviene

2. ogni occorrenza del taglio in una derivazione π può essere trasformata in una occorrenza *pronta* del taglio, cioé una occorrenza in cui le formule attive sono state appena introdotte. La trasformazione in questione è una banale applicazione della proprietà di *ascendenza* del taglio e verrà indicata col simbolo Ψ

3. ogni occorrenza pronta del taglio determina un caso-chiave di *riduzione del taglio*: la trasformazione Φ applicata ad un caso-chiave ha sempre l'effetto di diminuire la lunghezza delle formule attive (tali casi-chiave sono detti *logici*) o il peso del taglio (casi-chiave *strutturali*). La riduzione del taglio determina dei *residui*, che sono le occorrenze di taglio "figlie" generate dalla trasformazione Φ

4. poiché l'applicazione delle funzioni Ψ e Φ aumenta velocemente il numero totale di occorrenze di taglio in una derivazione logica, non è facile descrivere una strategia efficace per eliminare definitivamente tutti i tagli. Un uso accurato del rango e della coppia ordinata $\overrightarrow{v} = (\text{lunghezza}, \text{peso})$ di un taglio è tuttavia sufficiente alla cosa

5. a questo punto basta definire una trasformazione Υ il cui scopo sia quello di applicare Ψ e Φ ripetutamente ad una occorrenza di taglio fino a far diminuire strettamente il suo vettore $\overrightarrow{v}$ (ordinato lessicograficamente); per rendere questa procedura il più determinista possibile, bisogna dare un *ordine ciclico* sui tagli di rango zero ed applicare la trasformazione Υ agli elementi del ciclo nel loro ordine ciclico[2]. Il risultato sarà che ogni occorrenza di taglio di rango zero sarà eliminata: a questo punto, è sufficiente ripetere il procedimento sui tagli di rango uno e così via.

Il teorema di eliminazione del taglio è particolarmente utile nella logica intuizionista, come mostrato anche dall'esempio sottostante.

Esempio. *Se, per assurdo, fosse possibile derivare in* LJ *il principio del* tertium non datur *(ovvero il sequente:* $\vdash X \vee \neg X$*) dunque dovrebbe anche esistere una sua derivazione senza tagli nella quale* X *è una formula atomica generica; tuttavia l'unico modo di dimostrare ciò è usare una regola* $\vee^1_R$ *o* $\vee^2_R$ *ed entrambi i casi danno luogo ad una assurdità.*

Capitolo 4: Logica Lineare

Sintesi. *La* logica lineare *(o* LL*) fu fondata da J. Y. Girard nel 1987 [Gir87]: in questo calcolo logico si fa una severa distinzione tra i connettivi logici* moltiplicativi *(*$\otimes$*,* $\parr$*) e quelli* additivi *(*$\&$*,* $\oplus$*); inoltre, le regole strutturali sono limitate dalla introduzione di due operatori modali* ?*,* ! *(detti* esponenziali*). Introdurremo ora tale teoria e ne definiremo un frammento chiamato* logica lineare intuizionista *(o* ILL*).*

La logica lineare è definita su un linguaggio $\mathcal{L}_V$ di segnatura:

$$\mathcal{L}_V = (\mathcal{V}, \{\&, \oplus, \otimes, \parr, !, ?, \bigvee, \bigwedge, \top, \bot, \mathbf{0}, \mathbf{1}, (,)\}, \mathcal{C}, \mathcal{P}, \bigcup_{n\geq 1} \mathcal{R}_n, \bigcup_{n\geq 1} \mathcal{F}_n)$$

dove $\&$, $\otimes$ sono (rispettivamente) la congiunzione *additiva* e quella *moltiplicativa* mentre $\oplus$, $\parr$ sono (risp.) la disgiunzione *additiva* e quella *moltiplicativa*; i connettivi unari ! e ? sono detti *esponenziali*. La logica lineare ha a disposizione un quantificatore *esistenziale* $\bigvee$ (si legge *qualche*) ed uno *universale* $\bigwedge$ (si legge *qualsiasi*). Infine, in *LL* esistono quattro tipi distinti di *unità*: $\top$, $\mathbf{1}$ sono (risp.) la verità *additiva* e *moltiplicativa* mentre $\mathbf{0}$, $\bot$ sono la falsità *additiva* e *moltiplicativa*.

La negazione è definita direttamente sul linguaggio come nel calcolo dei sequenti: la tabella sotto mostra i rapporti di *dualità*. L'implicazione lineare $\multimap$ è invece definita nel seguente modo: $A \multimap B \stackrel{def}{=} A^{\perp} \parr B$.

[2] questa operazione, per essere eseguita correttamente, deve incrementare i posti del ciclo in modo da far posto agli eventuali residui di Υ: fortunatamente, con qualche accorgimento, è possibile eseguire questa operazione senza pericoli

$\diamondsuit$	$\diamondsuit^{\perp}$
$\top$	**0**
$\perp$	**1**
$\&$	$\oplus$
$\otimes$	$\parr$
$\bigwedge$	$\bigvee$
!	?

La logica lineare è stata alla base della nascita della teoria delle *prove di rete* (o *proof-nets*) ed è perciò il riferimento standard quando si parla di esse; tuttavia noi ci concentreremo prevalentemente su un determinato frammento di *LL*, ovvero quello *intuizionista* (o *ILL*).

Sebbene sarebbe stato possibile abbracciare le notazioni del frammento intuizionista *ILL* nel seguito di questa tesi, ciò si è rivelato piuttosto scomodo specie se in congiunzione con lo studio della regola dell'assurdo; inoltre la gestione degli esponenziali è stata ridotta all'osso e, di fatto, scansata attraverso la tecnica della ricostruzione delle *scatole* (i cosiddetti *box*): abbiamo ritenuto necessario, per questo motivo, introdurre un **nuovo calcolo logico** fortemente basato su *ILL* prima di introdurre infine la teoria delle reti di prova.

Capitolo 5: Calcolo Intuizionista Linearizzato

Sintesi. *In questo capitolo si definisce una variante del calcolo intuizionista che farà da base alla successiva definizione di rete di prova. Questo calcolo presenta numerosi legami con altri ben noti sistemi logici quali la* logica lineare intuizionista *(o* ILL*) e la* logica classica *(o* LC*, vedi [Gir91]).*

Il *calcolo intuizionista linearizzato* (o *LLJ*) è un calcolo logico basato su un linguaggio $\mathcal{L}_V$ con la seguente segnatura:

$$\mathcal{L}_V = (\mathcal{V}, \{\vee, \wedge, \to, \exists, \forall, \mathbb{T}, \mathbb{F}, (,)\}, \mathcal{C}, \mathcal{P}, \bigcup_{n\geq 1} \mathcal{R}_n, \bigcup_{n\geq 1} \mathcal{F}_n)$$

dove la negazione di una formula **senza occorrenze di implicazione** è costruita direttamente sul linguaggio.
Le regole di *LLJ* sono date in *tabella 3* (anche se, a volte, useremo le regole aggiuntive in *tabella 4*). Definiremo il concetto di *derivazione logica* in *LLJ* proprio come in logica intuizionista.

In un sequente linearizzato della forma $A_1, \ldots, A_n \vdash B$ diremo che gli A_i sono *entrate* mentre B è l'uscita. In generale, data una regola non-canonica $\odot$ allora denoteremo il calcolo LJ linearizzato con l'aggiunta della nuova regola attraverso la seguente notazione: $LLJ + \{\odot\}$.

Osservazione. Il calcolo logico *LLJ* ha la stessa potenza espressiva di *LK* dove si traduce un sequente classico $\vdash A_1, \ldots, A_n$ nel sequente linearizzato $A_n^{\perp}, \ldots, A_1^{\perp} \vdash$; allo stesso modo, $LLJ + \{\multimap, \lhd\}$ ha la stessa potenza espressiva di LJ privato delle regole sulla negazione.

Nota: nel seguito Ξ può essere vuoto o composto da un'unica formula

Identità $\dfrac{\emptyset}{A \vdash A}\, AX$ $\qquad$ $\dfrac{\emptyset}{A, A^{\perp} \vdash}\, LoC$ *Legge di Contraddizione* $\qquad$ $\dfrac{\Gamma \vdash A \qquad A, \Delta \vdash \Xi}{\Gamma, \Delta \vdash \Xi}\, CUT$

Moltiplicativi $\dfrac{\Gamma \vdash A_1 \qquad \Delta \vdash A_2}{\Gamma, \Delta \vdash A_1 \wedge A_2}\, \otimes$ $\qquad$ $\dfrac{\Sigma, A_1, A_2 \vdash \Xi}{\Sigma, A_1 \wedge A_2 \vdash \Xi}\, \parr$

Additivi $\dfrac{\Gamma, A_1 \vdash \Xi \qquad \Gamma, A_2 \vdash \Xi}{\Gamma, A_1 \vee A_2 \vdash \Xi}\, \&$ $\qquad$ $\dfrac{\Delta \vdash A_i}{\Delta \vdash A_1 \vee A_2}\, \oplus^i$

Quantificatori $\dfrac{\Gamma \vdash A(x)}{\Gamma \vdash \forall x : A(x)}\, \forall$ $\qquad$ $\dfrac{\Delta, A(\mathbf{t}/x) \vdash \Xi}{\Delta, \forall x : A(x) \vdash \Xi}\, \exists$

x non libera in Γ

$\dfrac{\Gamma, A(x) \vdash \Xi}{\Gamma, \exists x : A(x) \vdash \Xi}\, \bigwedge$ $\qquad$ $\dfrac{\Delta \vdash A(\mathbf{t}/x)}{\Delta \vdash \exists x : A(x)}\, \bigvee$

x non libera in $\Gamma \cup \Xi$

Unità $\dfrac{\emptyset}{\mathbb{F} \vdash}\, \top$ $\qquad$ $\dfrac{\emptyset}{\vdash \mathbb{T}}\, \mathbf{1}$

$\dfrac{\Gamma \vdash \Xi}{\Gamma, \mathbb{T} \vdash \Xi}\, \bot$ $\qquad$ $\dfrac{\Gamma \vdash}{\Gamma \vdash \mathbb{F}}\, \mathbf{0}$

Strutturali $\dfrac{\Gamma, A, A \vdash \Xi}{\Gamma, A \vdash \Xi}\, C$ $\qquad$ $\dfrac{\Gamma \vdash \Xi}{\Gamma, A \vdash \Xi}\, \uparrow W$ $\qquad$ $\dfrac{\Gamma \vdash}{\Gamma \vdash A}\, \downarrow W$ *Legge di Esplosione*

Tabella 3: Regole di *LLJ*

La regola R, o *reductio ad absurdum*, è molto potente: si può dimostrare che essa è equivalente al *tertium non datur* e, inoltre, permette di dimostrare la legge di contrapposizione, l'equivalenza disgiunzione/implicazione, la legge di Peirce ed il principio del bevitore.

Capitolo 6: l'*Hauptsatz* nella Logica Intuizionista Linearizzata

Sintesi. *Mostreremo in questo capitolo la teoria della eliminazione del taglio per derivazioni in* $\mathrm{LLJ} + \{R\}$ *e mostreremo infine la consistenza sintattica di tale calcolo. La prova di questi risultati, da un punto di vista della teoria della dimostrazione, giustifica l'uso matematico della regola di riduzione all'assurdo.*

Per dimostrare l'eliminazione del taglio in $LLJ + \{R\}$ è sufficiente riportarsi alla dimostrazione in LJ, facendo attenzione però ad alcune differenze significative:

Reductio ad absurdum	$\dfrac{\Gamma, A \vdash}{\Gamma \vdash A^{\perp}}\, R$ A senza occorrenze di '$\rightarrow$'
Implicazioni	$\dfrac{\Gamma, A \vdash B}{\Gamma \vdash A \rightarrow B} \multimap$ $\qquad$ $\dfrac{\Delta \vdash A \qquad \Sigma, B \vdash \Xi}{\Delta, \Sigma, A \rightarrow B \vdash \Xi} \lhd$

Tabella 4: Regole non-canoniche di *LLJ*

1. una occorrenza della regola R è *pronta* sse la sua formula attiva è stata appena introdotta; in generale, è sempre possibile trasformare una occorrenza della regola R in una occorrenza pronta purché si escludano dal linguaggio le due regole della implicazione $\multimap$ e $\lhd$
2. un taglio contro una regola dell'assurdo è *pronto* sse anche la regola R è pronta; in tal caso, ci si ritrova nuovamente in una lista di casi chiave di tipo logico o di tipo strutturale
3. le nozioni di *rango*, *peso* e *lunghezza* rimangono immutate e lo stesso vale per il procedimento della dimostrazione dell'*Hauptsatz*.

Il teorema di eliminazione del taglio in $LLJ + \{R\}$ permette di dimostrare la consistenza sintattica del calcolo logico e l'indipendenza della regola dell'assurdo dalle altre regole di *LLJ*. Non solo: è anche possibile dimostrare che ogni derivazione del calcolo può essere trasformata in una nuova derivazione in cui compare un'unica occorrenza di R e tale occorrenza è anche l'ultima regola.

Capitolo 7: Grafi e Reti di Prova

Sintesi. *Dopo aver mostrato brevemente la* teoria dei grafi*, useremo le sue nozioni per introdurre le* reti di prova *(in inglese,* proof-nets*) [Gir96]. La teoria delle reti di prova è uno dei risultati più avanzati delle ultime decadi in logica matematica: l'idea di fondo è quella di costruire un oggetto geometrico che identifichi due deduzioni che differiscono unicamente nell'ordine di commutazione delle sue regole. In particolare, vedremo quali sono le proprietà particolari delle reti di prova in* LLJ $+ \{R, \multimap, \lhd\}$ *e daremo un algoritmo che permetterà di ricostruire le* scatole *senza bisogno di dichiararle tramite regole esponenziali.*

Una rete di prova $(G, \mathbf{L}, \mathbf{L}')$ è una tripla composta da:

1. un grafo $G = (V, E)$ dove V è l'insieme delle sue celle[3] ed E è l'insieme dei suoi archi

[3] una cella, per dirlo in maniera informale, è un tipo di vertice che distingue l'ordine di inserimento dei suoi archi

2. la funzione $\mathbf{L} : V \mapsto X_V$ che associa ad ogni cella un 'label' corrispondente ad una regola di $LLJ + \{R, \multimap, \lhd\}$

3. la funzione $\mathbf{L}' : E \mapsto X_E$ che associa ad ogni arco un 'label' corrispondente ad una formula di $\mathcal{L}_V$.

Il grafo G sarà sempre dotato di un unico arco terminale uscente, chiamato appunto *uscita* del grafo.

Definizione (Rete di prova)**.** La funzione $\Theta : \Pi \mapsto \ulcorner\Pi\urcorner$ (dove Π è l'insieme delle derivazioni nel calcolo $LLJ + \{R, \multimap, \lhd\}$ e $\ulcorner\Pi\urcorner$ è l'insieme dei grafi con labels) è la funzione definita per induzione sulla complessità di una derivazione π nel seguente modo:

- ad ogni regola iniziale di n entrate ed una uscita (anche vuota) corrisponde una cella collegata ad n archi entranti ed un arco uscente; il label della cella è il nome della regola ed i label degli archi sono quelli delle formule

- ad ogni regola non iniziale si associa una cella collegata agli archi corrispondenti alle sue occorrenze di formule attive; se la regola introduce una formula-entrata (risp. uscita) allora si crea un nuovo arco entrante nella (risp. uscente dalla) cella

- la regola $\uparrow W$ e la regola & sono due casi particolari: nel primo caso, creeremo un arco speciale aggiuntivo detto *salto* (in inglese, *jump*) il cui scopo è quello di connettere la cella $\uparrow W$ alla cella da cui parte l'uscita del grafo (tale cella è sempre univocamente determinata); nel secondo caso creeremo una cosiddetta *scatola additiva* (in inglese, *additive box*), ovvero delimiteremo con celle speciali (dette *porte* o *doors*) le parti del grafo che corrispondono alle due derivazioni-ipotesi della regola &. Per la costruzione esplicita di una scatola additiva, si rimanda a [Gir96].

L'immagine di Θ è un grafo con labels $(G, \mathbf{L}, \mathbf{L}')$ che sarà detto *rete di prova*.

Osservazione. L'uso delle scatole additive si rifà alla definizione girardiana originale; vogliamo far notare, tuttavia, che di recente sono state elaborate delle tecniche più sofisticate basate sul concetto di *slices* [LF04].

Un arco α di una rete di prova può essere collegato, al più, a due celle: diremo che α è *up* rispetto ad una di esse ed è *down* rispetto all'altra come mostrato in *tabella 5*. Diremo anche che un arco è una *entrata locale* (risp. *uscita locale*) sse è *entrante* (risp. *uscente*) rispetto alla cella di cui è un arco *down*: si noti che ogni arco è una entrata locale o una uscita locale, ma mai entrambe.

Proposizione. *Se due derivazioni prive di occorrenze della regola & sono equivalenti modulo commutazioni di regole allora esse formano la stessa rete di prova (escludendo gli archi che rappresentano dei* salti*).*

Questa proposizione rafforza l'intuizione che una rete di prova catturi "l'essenza" di una derivazione, rendendo obsolete certe proprietà quali la *ascendenza/discendenza* di una regola o la definizione di taglio *pronto*. Inoltre, le commutazioni di regole sono uno dei fattori più rilevanti a livello computazionale nel processo di eliminazione del taglio [Ore79]: poterne fare a meno è uno dei grandi meriti delle reti di prova.

Definizione. Un dato percorso in una rete di prova è detto *percorso regolare* sse è orientato, privo di archi ripetuti e la sua fine corrisponde all'uscita del grafo.

Proposizione. *Ogni arco di una rete di prova è collegato alla sua uscita da un* percorso regolare.

Il fatto che ogni arco sia collegato alla uscita del grafo tramite un percorso regolare è alla base della tecnica che permette di ricostruire una **scatola** ovvero un sottografo che sia, a sua volta, una rete di prova: l'idea di base è quella di esplorare il grafo seguendo tutti (e soli) i percorsi anti-orientati! Poter ricostruire una scatola è essenziale per definire correttamente la riduzione del taglio sulle reti di prova: i casi strutturali richiedono sempre, infatti, di eliminare/duplicare una parte del grafo. Mentre in logica lineare l'inscatolamento viene dichiarato ufficialmente attraverso l'esponenziale '!', in questo lavoro abbiamo definito invece una ricostruzione implicita delle scatole.

Definizione (Distanza di un arco)**.** Dato un percorso regolare $\maltese$ in una rete di prova $(G, \mathbf{L}, \mathbf{L}')$ e dato un arco α di $\maltese$, definiremo come *distanza* il numero totale di occorrenze distinte di uscite locali contenute in $\maltese$ tra l'inizio del percorso ed α (inclusi) e la denoteremo tramite $\mathbf{d}^G_{\maltese}(\alpha)$.

Dato un arco α in una rete di prova $(G, \mathbf{L}, \mathbf{L}')$, diremo che la *distanza di* α *in* G è la sua massima distanza tra tutti i possibili percorsi regolari $\maltese$ che contengono α e la denoteremo tramite $\mathbf{d}^G(\alpha)$:

$$\mathbf{d}^G(\alpha) \stackrel{def}{=} \max\{\mathbf{d}^G_{\maltese}(\alpha) \mid \maltese \text{ percorso regolare contenente } \alpha\}$$

Definizione (Scatola)**.** Data una uscita locale α in una rete di prova, si definisce la sua scatola per induzione sulla distanza di α:

- se α è uscente rispetto la cella $\odot$, allora la scatola *temporanea* di α sarà l'unione dell'arco α, della cella $\odot$ e delle scatole delle uscite locali entranti in $\odot$ (che, per costruzione, hanno una *distanza* strettamente minore)

- una scatola temporanea diventa *completa* inglobando il più possibile tutte le celle esterne del tipo $\parr$, $\exists$, $\bigwedge$, $\uparrow W$ e C i cui archi uscenti sono nella scatola (questa operazione è detta *completamento grezzo*)

ed inglobando il più possibile tutte le celle esterne del tipo CUT e $\lhd$ il cui arco uscente è nella scatola; in questi ultimi due casi, ingloberemo anche le scatole costruite sulle uscite locali collegate alla cella (tale operazione è detta *completamento pieno*).

Si noti che la scatola definita in questo modo è *massimale* nel senso che la rete di prova di una sottoderivazione qualsiasi avente α come uscita è sempre contenuta nella scatola.

Osservazione. Nella definizione data, una scatola è sempre costruita su una **uscita locale**: sebbene ciò sia sufficiente per definire la riduzione del taglio nelle reti di prova di un frammento limitato del calcolo[4], purtroppo non è ancora abbastanza per definire la riduzione del taglio in presenza di scatole additive[5]. Uno degli obiettivi futuri è definire la scatola costruita su una **entrata locale**, possibilmente sfruttando le *slices* in luogo delle scatole additive.

Capitolo 8: Eliminazione del Taglio su Reti di Prova

Sintesi. *Abbiamo visto come introdurre un oggetto matematico, la* rete di prova*, che cattura il concetto di "parallelismo" di una deduzione nella logica intuizionista linearizzata (escludendo scatole additive e salti). Mostreremo ora che le reti di prova sono suscettibili di* eliminazione del taglio *e che tale risultato è raggiungibile attraverso degli strumenti* ***puramente geometrici****.*

La teoria delle reti di prova in logica lineare consente di dimostrare la commutatività del diagramma che lega le riduzioni del taglio tramite sequenti alle corrispondenti riduzioni del taglio tramite reti di prova; per fare ciò, l'uso delle scatole è essenziale: tuttavia l'inscatolamento in una rete di prova può essere dichiarato in più modi equivalenti, e solo uno di essi consente al diagramma di commutare. La ricostruzione implicita delle scatole in $LLJ + \{R, \multimap, \lhd\}$ ha dunque il ragguardevole difetto che, in generale, il diagramma mostrato sotto non commuta.

Per ridimostrare l'eliminazione del taglio useremo perciò il seguente procedimento: definiremo su ogni occorrenza di taglio nella rete di prova $\Theta(\pi)$ un

[4] ovvero il frammento composto da AX, $\mathbf{1}$, $\top$, CUT, $\otimes$, $\parr$, $\multimap$, $\forall$, $\exists$, $\lhd$, $\uparrow W$ e C

[5] ad esempio, non è possibile definire la riduzione di taglio contro l'uscita di una scatola additiva

"*valore*" in modo che, se π è equivalente a π' modulo commutazioni di regole, allora il valore di una cella-CUT rimane invariato; inoltre, ogni caso-chiave di riduzione del taglio diminuirà strettamente tale valore. La cosa veramente degna di nota è che tale *valore* è definito tramite **proprietà puramente geometriche** della rete di prova, cioé attraverso il conteggio di occorrenze di celle ed archi lungo un cammino regolare.

Definizione. Il *peso strutturale* di un arco in una rete di prova è il massimo numero di occorrenze di contrazioni in un percorso regolare (tra l'arco dato e la fine del percorso) calcolato tra tutti i percorsi regolari passanti per quel arco.

Il *peso strutturale* di una occorrenza di cella-CUT è il peso strutturale del suo arco uscente; la *distanza* di una occorrenza di cella-CUT è la distanza del suo arco entrante; il *valore* di una occorrenza di cella-CUT è la somma del suo peso strutturale e la sua distanza.

Definizione. Diremo che un caso-chiave di riduzione del taglio in una rete di prova *riduce la distanza* sse la distanza dei suoi residui è stata ridotta, il peso strutturale è rimasto invariato e nessun'altro *valore* di un CUT è stato aumentato.

É facile mostrare che tutti i casi-chiave logici in LLJ corrispondono a riduzioni del taglio che riducono la distanza; viceversa, i casi-chiave strutturali corrispondono a riduzioni di taglio che riducono il peso strutturale: in ogni caso, l'applicazione di una riduzione di taglio diminuisce strettamente il *valore* dei suoi residui senza mai alterare il valore delle altre celle-CUT.

Osservazione. Mentre tutti i casi-chiave logici in LLJ sono riduzioni di taglio che *riducono la distanza*, ciò non è più vero in $LLJ + \{\multimap, \triangleleft\}$ e neppure in $LLJ + \{R\}$: casi-chiave come $\multimap \ / \triangleleft$ oppure $\parr + R \ / \ \&$ hanno la peculiarità di modificare i percorsi regolari, allungandoli a piacimento (vedi anche in *tabella 6*). Il problema sembra giacere nel fatto che tali celle sono le uniche che introducono cicli orientati all'interno della rete di prova.

Teorema (*Hauptsatz* geometrico)**.** *Ogni derivazione di* LLJ *può essere trasformata in una derivazione (avente la stessa conclusione) priva di occorrenze di regole-CUT.*

La dimostrazione di questo teorema si basa sull'*argomento dell'idra*: se la riduzione di un taglio genera un numero di residui a piacere, ma tali residui hanno tutti *valore* strettamente inferiore, allora qualsiasi strategia di riduzione del taglio è, definitivamente, un processo di eliminazione del taglio.

Il fatto che si possa dimostrare l'eliminazione del taglio in maniera così generale costituisce un buon terreno di studio per stabilire quali regole della logica matematica creino problemi (ad esempio per quali regole le reti di prova non godano della eliminazione del taglio o, peggio, per quali regole le

reti di prova possano generare antinomie!). A questo scopo, introdurremo nel prossimo capitolo gli assiomi di comprensione ed alcune definizioni di uguaglianza.

Capitolo 9: Paradossi e Assiomi di Comprensione

Sintesi. *In questo capitolo arricchiremo il nostro calcolo con regole di inferenza classiche della teoria naïve degli insiemi: gli* assiomi di comprensione. *Grazie ad essi mostreremo e commenteremo alcuni dei risultati più noti in logica matematica; in particolare, il* paradosso di Curry *sarà utile per "tipare" certe derivazioni tramite generici λ-termini.*

Come è noto, la teoria naïve degli insiemi è inconsistente nel senso che al suo interno è possibile generare antinomie: cosa si può dire dunque del calcolo logico *LLJ* quando aggiungiamo i seguenti *assiomi di comprensione*?

$$\frac{\Gamma \vdash A(\mathbf{t}/x)}{\Gamma \vdash \mathbf{t} \in \{x \mid A(x)\}} \downarrow \lambda \qquad \frac{\Delta, A(\mathbf{t}/x) \vdash \Xi}{\Delta, \mathbf{t} \in \{x \mid A(x)\} \vdash \Xi} \uparrow \lambda^{\perp}$$

$$\frac{\Gamma \vdash A(\mathbf{t}/x)}{\Gamma \vdash \mathbf{t} \notin \{x \mid A(x)^{\perp}\}} \downarrow \lambda^{\perp} \qquad \frac{\Delta, A(\mathbf{t}/x) \vdash \Xi}{\Delta, \mathbf{t} \notin \{x \mid A(x)^{\perp}\} \vdash \Xi} \uparrow \lambda$$

Definizione (Calcolo delle classi)**.** Dato il seguente linguaggio $\mathcal{L}_V$:

$$\mathcal{L}_V = (\mathcal{V}, \{\vee, \wedge, \in, \notin, \exists, \forall, \mathbb{T}, \mathbb{F}, (,)\}, \mathcal{C}, \mathcal{P}, \bigcup_{n \geq 1} \mathcal{R}_n, \bigcup_{n \geq 1} \mathcal{F}_n)$$

definiremo *calcolo delle classi* il calcolo logico *LLJ* munito di assiomi di comprensione.

All'interno del calcolo delle classi è possibile definire concetti insiemistici quali le *unioni/intersezioni* di classi, le *famiglie*, la *classe potenza*, la *classe vuota/classe universo*, l'identità *estensionale*, l'identità *di Leibniz* e via dicendo. Nel corso del capitolo rivedremo alcuni risultati classici della teoria degli insiemi, quali il teorema di Cantor, il paradosso di Russell e quello di Curry.

Consistenza sintattica Il calcolo delle classi è sintatticamente consistente in quanto è possibile applicare in esso la dimostrazione dell'*Hauptsatz* geometrico. Si noti che per il calcolo delle classi non è possibile dare una dimostrazione dell'*Hauptsatz* di tipo "classico": infatti, in presenza degli assiomi di comprensione, non è neppure possibile dire se una formula è sottoformula di un'altra[6] né, tantomeno, parlare di *lunghezza* di una formula.

[6] come nel caso delle due formule $\mathbf{u} \in \mathbf{u}$ ed $\mathbf{u} \notin \mathbf{u}$ dove: $\mathbf{u} \stackrel{def}{=} \{x \mid x \notin x\}$

Paradosso di Russell Il paradosso di Russell, nella teoria naïve degli insiemi, è una vera e propria antinomia: nel calcolo delle classi invece, a causa della consistenza sintattica, il paradosso di Russell si trasforma nella dimostrazione del sequente:

$$\mathbf{u} \in \mathbf{u} \vee \mathbf{u} \notin \mathbf{u} \vdash \qquad \text{dove: } \mathbf{u} \stackrel{def}{=} \{x \,|\, x \notin x\}$$

Ne consegue che nel calcolo delle classi non vale il principio del *tertium non datur*[7]: poiché tale principio in *LLJ* è equivalente ad accettare la *reductio ad absurdum*, dunque il paradosso di Russell afferma che il calcolo delle classi munito di riduzione all'assurdo è sintatticamente inconsistente.

Paradosso di Curry Il paradosso di Curry è la più importante antinomia in deduzione naturale con assiomi di comprensione: il paradosso, applicato al calcolo delle classi munito di implicazione, consiste della seguente dimostrazione

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\mathbf{z} \in \mathbf{z} \vdash \mathbf{z} \in \mathbf{z} \qquad Z \vdash Z}{\mathbf{z} \in \mathbf{z}, (\mathbf{z} \in \mathbf{z}) \to Z \vdash Z}\,\lhd}{\mathbf{z} \in \mathbf{z}, \mathbf{z} \in \{x \,|\, (x \in x) \to Z\} \vdash Z}\,\uparrow \lambda^{\perp}}{\mathbf{z} \in \mathbf{z} \vdash Z}\,C}{\vdash (\mathbf{z} \in \mathbf{z}) \to Z}\,\multimap}{\vdash \mathbf{z} \in \{x \,|\, (x \in x) \to Z\}}\,\downarrow \lambda \qquad \dfrac{\dfrac{\dfrac{\dfrac{\mathbf{z} \in \mathbf{z} \vdash \mathbf{z} \in \mathbf{z} \qquad Z \vdash Z}{\mathbf{z} \in \mathbf{z}, (\mathbf{z} \in \mathbf{z}) \to Z \vdash Z}\,\lhd}{\mathbf{z} \in \mathbf{z}, \mathbf{z} \in \{x \,|\, (x \in x) \to Z\} \vdash Z}\,\uparrow \lambda^{\perp}}{\mathbf{z} \in \mathbf{z} \vdash Z}\,C}{\vdash Z}\,CUT$$

dove Z è una formula qualsiasi e: $\mathbf{z} = \{x \,|\, (x \in x) \to Z\}$. In particolare, il paradosso di Curry afferma che il calcolo delle classi munito di implicazione è sintatticamente inconsistente.

Osservazione. Si noti che tale derivazione è un punto fisso rispetto al processo di eliminazione del taglio: questa caratteristica sarà giustificata nel prossimo capitolo attraverso una operazione di "tipaggio" della prova nel λ-calcolo.

Capitolo 10: Il λ-Calcolo

Sintesi. *Il λ-calcolo fu introdotto intorno al* 1930 *da A. Church come parte di uno studio riguardante i fondamenti della matematica, sebbene il formalismo usato fosse stato introdotto precedentemente da M. I. Schöfinkel e H. Curry: il calcolo fuoriuscito era così espressivo che poteva essere utilizzato perfino come definizione alternativa della famiglia delle* funzioni ricorsive. *Ulteriori indagini mostrarono come il λ-calcolo, con alcune implementazioni aggiuntive, potesse essere usato quale efficace* linguaggio di programmazione funzionale: *il* LISP *di J. McCarthy, sviluppato nel 1959 [McC59], ne divenne un esempio pratico mentre P. J. Landin [Lan66], J. C. Reynolds [Rey83] e J. Y. Girard [GLT89] ne studiarono varie potenzialità interessanti.*

[7] di conseguenza, il calcolo delle classi è *inconsistente* pur essendo *sintatticamente consistente*!

Definizione (λ-calcolo)**.** La sintassi del λ-calcolo è la seguente:

$$M ::= x \parallel M\,M \parallel \lambda x.M$$

dove x è una *variabile*, $M_1\,M_2$ è una *applicazione* e $\lambda x.M$ è una *astrazione*. L'insieme di tutti i λ-termini è indicato da Λ. L'insieme numerabile di tutte le variabili è indicato da X.

Le seguenti sono abbreviazioni o termini molto comuni del λ-calcolo.

$$\lambda x_1 \dots x_n.M = \lambda x_1.(\dots \lambda x_n.M \dots) \qquad MN_1 \dots N_n = (\dots (MN_1) \dots N_n)$$
$$\mathbf{Id} \stackrel{\text{def}}{=} \lambda x.x \qquad \mathbf{K} \stackrel{\text{def}}{=} \lambda xy.x$$
$$\Delta \stackrel{\text{def}}{=} \lambda x.xx \qquad \mathbf{S} \stackrel{\text{def}}{=} \lambda xyz.(xz)(yz)$$
$$\mathbf{Y} \stackrel{\text{def}}{=} \lambda f.(\lambda x.f(xx))(\lambda x.f(xx))$$

dove **Id** è l'*identità*, **S** e **K** sono i *combinatori SKI*, Δ è il termine che si *applica a se stesso* e **Y** è il combinatore punto-fisso di *Haskell-Curry*.

In una astrazione $\lambda x.\mathbf{t}$ si dice che la variabile x è quantificata: in generale, si definisce una α-equivalenza tra λ-termini che differiscono a meno di variabili quantificate e si definisce una *sostituzione sintattica* tra λ-termini in modo che le variabili quantificate siano ignorate nel procedimento di sostituzione. Inoltre, si definisce la operazione di β-riduzione come:

$$(\lambda x.\mathbf{t})\,\mathbf{u} \;\leadsto_\beta\; \mathbf{t}\,[\mathbf{u}/x]$$

e si definisce la β-equivalenza tra λ-termini che differiscono a meno di β-riduzioni di sottotermini.

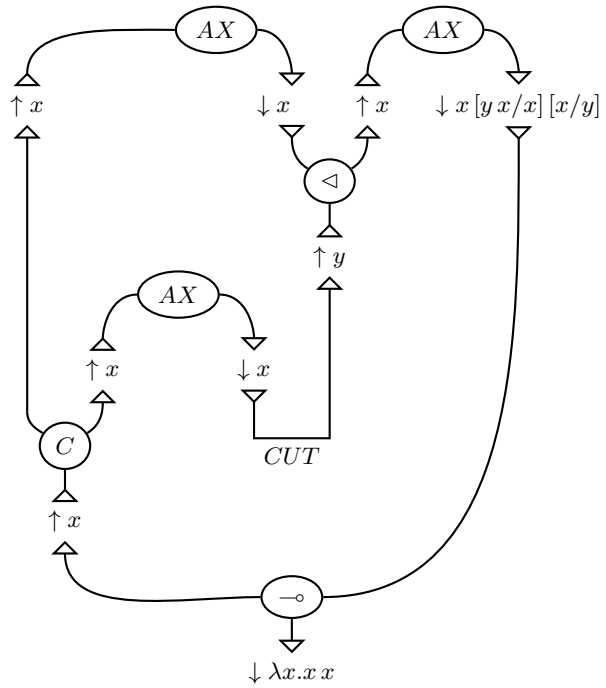
Per finire, si definisce una *derivazione* $\mathbf{t} \leadsto^* \mathbf{t}'$ come una sequenza data di passi di β-riduzione che portano dal λ-termine $\mathbf{t}$ al λ-termine $\mathbf{t}'$.

Teorema (Proprietà di *Church-Rosser*)**.** *Se* $\mathbf{t} \leadsto^* \mathbf{u}$ *e* $\mathbf{t} \leadsto^* \mathbf{v}$ *allora esiste un* λ*-termine* $\mathbf{z}$ *tale che:* $\mathbf{u} \leadsto^* \mathbf{z}$ *e* $\mathbf{v} \leadsto^* \mathbf{z}$*,* cioé *le* β*-derivazioni sono* globalmente confluenti.

Per la proprietà di Church-Rosser, se una derivazione termina definitivamente dunque il λ-termine ottenuto è *unico* e viene detto essere in *forma normale*. In generale, non tutte le derivazioni terminano: ad esempio $\Delta\,\Delta$ si riduce sempre in se stesso.
Se definiamo la *derivazione normale* come la derivazione che riduce sempre i sottotermini più a sinistra, dunque si ha la seguente:

Teorema. *Se un* λ*-termine è normalizzabile dunque la sua derivazione normale termina.*

Uno dei grandi risultati ottenuti grazie allo studio del λ-calcolo è la cosiddetta *corrispondenza di Curry-Howard*, che lega un λ-termine *tipato* ad una derivazione di un frammento della *deduzione naturale intuizionista* (o NJ). Quel che faremo in questo capitolo è ricomporre tale relazione nel frammento di $LLJ + \{\multimap, \triangleleft\}$ composto da: AX, CUT, $\multimap$, $\triangleleft$, $\uparrow W$, C, $\downarrow \lambda$ e $\uparrow \lambda^{\perp}$ allo scopo di verificare se la corrispondenza continui a sussistere: sfortunatamente, tale relazione allo stato attuale può solo essere congetturata. Quel che è possibile fare è definire una rete di prova per un λ-termine in modo molto simile a quanto fatto da O. Laurent sulle reti di prova *MELL* polarizzate [Lau08]: nel caso del λ-termine Δ troviamo la seguente rete



che risulta anche essere quasi identica[8] alla rete di prova della seguente derivazione di tipo:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{x : \mathbf{z} \in \mathbf{z} \vdash x : \mathbf{z} \in \mathbf{z} \qquad \dfrac{x : \mathbf{z} \in \mathbf{z} \vdash x : \mathbf{z} \in \mathbf{z} \qquad x : \mathbf{z} \in \mathbf{z} \vdash x : \mathbf{z} \in \mathbf{z}}{x : \mathbf{z} \in \mathbf{z}, y : (\mathbf{z} \in \mathbf{z}) \to (\mathbf{z} \in \mathbf{z}) \vdash x\,[y\,x/x] : \mathbf{z} \in \mathbf{z}}\,\triangleleft}{x : \mathbf{z} \in \mathbf{z}, y : \mathbf{z} \in \mathbf{z} \vdash y\,x : \mathbf{z} \in \mathbf{z}}\,\uparrow\lambda^{\perp}}{x : \mathbf{z} \in \mathbf{z}, x : \mathbf{z} \in \mathbf{z} \vdash y\,x\,[x/y] : \mathbf{z} \in \mathbf{z}}\,CUT}{x : \mathbf{z} \in \mathbf{z} \vdash x\,x\,[x/x,\, x/x] : \mathbf{z} \in \mathbf{z}}\,C}{\vdash \lambda x.x\,x : (\mathbf{z} \in \mathbf{z}) \to (\mathbf{z} \in \mathbf{z})}\,\multimap}{\vdash \lambda x.x\,x : \mathbf{z} \in \mathbf{z}}\,\downarrow\lambda$$

derivazione che, infine, permette di risalire al paradosso di Curry[9] dove:

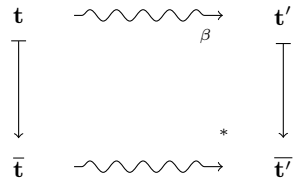
$$\mathbf{z} \stackrel{def}{=} \{x \,|\, (x \in x) \to (x \in x)\}$$

[8] la differenza è che le regole $\downarrow \lambda$, $\uparrow \lambda^{\perp}$ non vengono disegnate ed i label degli archi sono i tipi invece delle formule

[9] per essere precisi, il paradosso di Curry corrisponde al λ-termine $\Delta\,\Delta$

Questo comportamento ci suggerisce una ulteriore congettura: se $\overline{\mathbf{t}}$ è la rete di prova corrispondente al λ-termine $\mathbf{t}$, dunque:

Congettura. Il seguente diagramma è commutativo



Si noti che il nostro calcolo non ha operatori esponenziali, quindi le *scatole* di $\overline{\mathbf{t}}$ non vengono dichiarate poiché sono sempre ricostruibili implicitamente.

Osservazione. Se la precedente congettura è vera, dunque una derivazione normale corrisponde, nel diagramma, alle riduzioni di taglio delle celle-CUT di massima distanza.

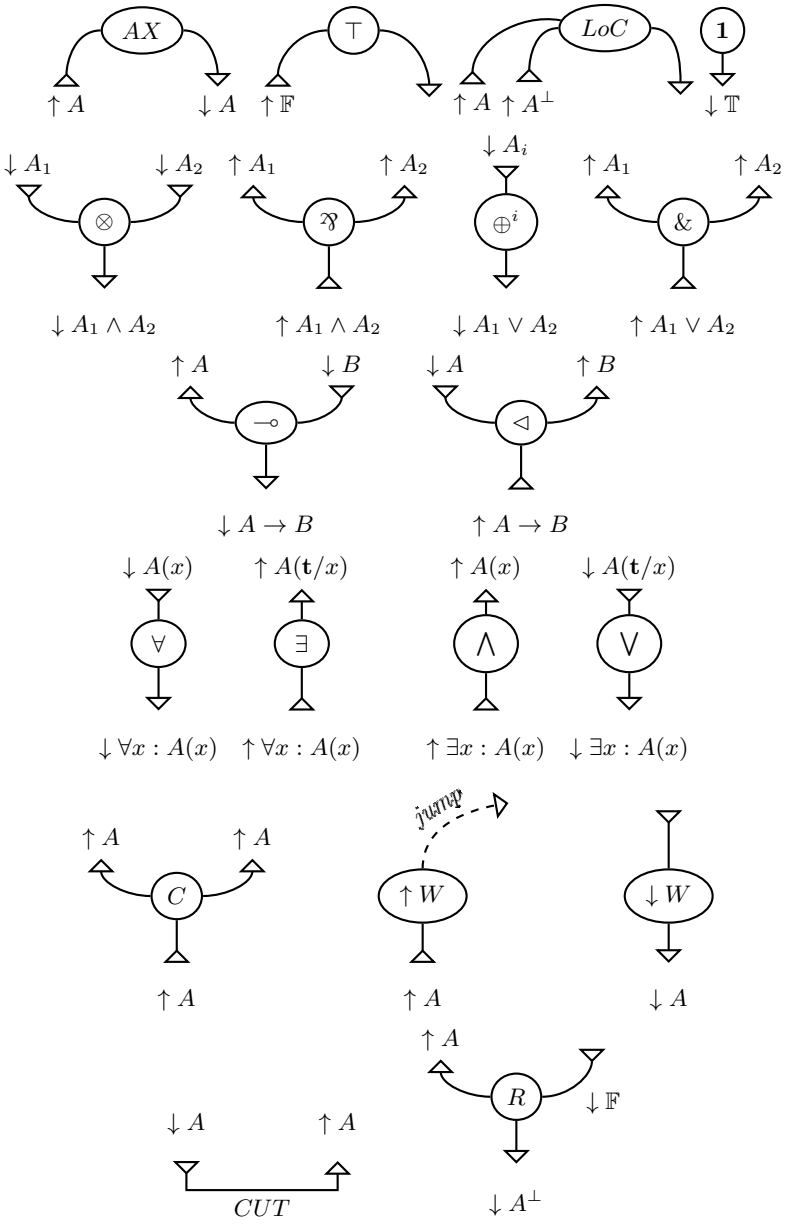
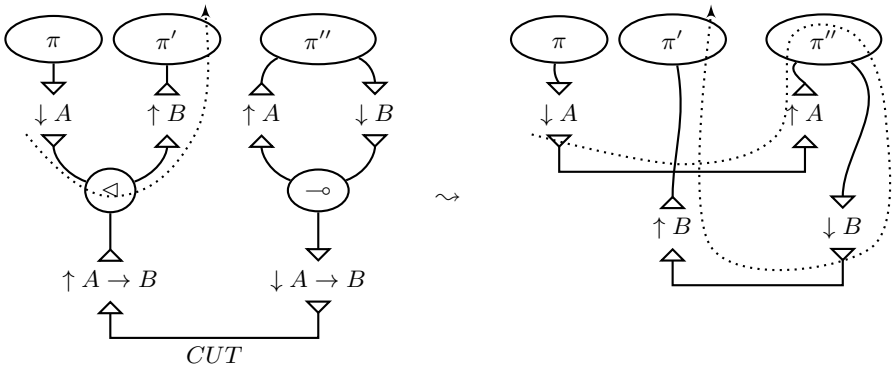


Tabella 5: Le celle di una rete di prova



Nota che il percorso punteggiato viene trasformato in un percorso più lungo passante attraverso la prova π'', *cioé* la prova che formava un *ciclo orientato* con la cella 

Tabella 6: Un esempio di riduzione del taglio che incrementa le *distanze*

Bibliografia

[AF09] Vito Michele Abrusci e Lorenzo Tortora de Falco. *Appunti del corso di Logica*. 2009.

[GLT89] Jean-Yves Girard, Yves Lafont e Paul Taylor. *Proof and Types*. Cambridge Tracts in Theoretical Computer Science 7. Cambridge University Press, 1989.

[Gen34] Gerhard Karl Erich Gentzen. "Untersuchungen über das logische Schließen". In: *Mathematische Zeitschrift 39. English translation* (1934).

[Gir87] Jean-Yves Girard. "Linear Logic". In: *Theoretical Computer Science* 50 (1987), pp. 1–102.

[Gir91] Jean-Yves Girard. "A New Constructive Logic: Classical Logic". In: *Mathematical Structures in Computer Science* 1.3 (1991), pp. 255–296.

[Gir96] Jean-Yves Girard. "Proof-nets: The parallel syntax for proof-theory". In: *Logic and Algebra*. Marcel Dekker, 1996, pp. 97–124.

[LF04] Olivier Laurent e Lorenzo Tortora De Falco. "Slicing polarized additive normalization". In: *Linear Logic in Computer Science*. A cura di Thomas Ehrhard et al. London Mathematical Society Lecture Note Series. Cambridge University Press, 2004, pp. 247–282.

[Lan66] Peter J. Landin. "The next 700 programming languages". In: *Communications of the ACM* 9 (1966), pp. 157–166. ISSN: 0001-0782.

[Lau08] Olivier Laurent. *Théorie de la démonstration*. Master de Logique Mathématique et Fondements de l'Informatique. 2008. URL: `http://perso.ens-lyon.fr/olivier.laurent/thdem.pdf`.

[McC59] John McCarthy. *Recursive Functions of Symbolic Expressions and their Computation by Machine*. Rapp. tecn. Cambridge, MA, USA: MIT, 1959.

[Ore79] V. P. Orevkov. "Lower bounds for the lengthening of proofs after cut-elimination". In: *Zapiski Nauchnykh Seminarov Leningradskogo Otdelenyia Ordena Lenina Matematicheskogo Instituta imeni Steklova Akademii Nauk SSSR (LOMI), 88* (1979), pp. 137–162.

[Pra65] Dag Prawitz. *Natural Deduction: A proof-theoretical study*. Stockholm: Almqvist e Wiksell, 1965.

[Rey83] John C. Reynolds. "Types, Abstraction, and Parametric Polymorphism". In: *Information Processing 83, Paris, France*. A cura di R. E. A. Mason. Elsevier, 1983, pp. 513–523.