# Metodi Formali dell'Informatica

Modulo A — Logica Matematica

vedasi mia home page per ogni info (appelli, etc.)

Salvatore Caporaso
Dipartimento di Informatica dell'Università di Bari
caporaso@di.uniba.it

November 8, 2008

# Contents (in progress)

# Chapter 1

# Elementi di analisi logica e di meta-logica

Qesto capitolo presenta alcune ipotesi introduttive sulla logica matematica. A carattere pre-matematico e senza pretese di statuto filosofico, esse si collocano in un limbo intermedio tra le due discipline. Sono qui esposte in forma estremamente schematica, per facilitare una prima intuizione della natura degli enti e strumenti che stiamo per analizzare.

## 1.1 Elementi di analisi logica

**Classi** Una classe è una raccolta di oggetti del pensiero o dell'intuizione. Se un oggetto $O$ fa parte di una classe $C$, si dice che *$O$ è in $C$*. Due classi sono uguali se e solo se sono costituite dagli stessi oggetti, o, come anche si dice, se hanno la stessa *estensione*.
In matematica, alle nozioni di *classe* e di *essere in* corrispondono, con certe cautele tecniche, le definizioni di *insieme* e di *appartenenza*. Una volta date queste definizioni, si può riuscire a *dimostrare* che la classe $C$ è un insieme (e anche no — ci sono classi che *non* sono insiemi) e che $O$ gli appartiene: in tal caso si scrive $O \in C$.

**Senso e significato** Un nome proprio può essere semplice o composto, ma la distinzione è spesso inessenziale, come accade per esempio con "il figlio di Atreo" e "Atride" o con "la storia di Ulisse" e "Odissea": in molte lingue i nomi propri composti sono spesso riconoscibili perché usati con l'articolo determinativo.
È piú importante osservare che un nome proprio spesso induce due atti mentali distinti: un concetto o una descrizione vengono *capiti* e, nel contempo, una singola entità, indicata da quel nome, viene *individuata*. Si può cogliere questa distinzione dicendo che un nome

proprio *esprime un senso*, che dovrebbe essere comprensibile a tutti coloro che intendono la particolare lingua in cui è formulato; e che, d'altra parte, esso *denota un significato*. Naturalmente può accadere che un nome, pur avendo un senso preciso, denoti un significato immaginario, o addirittura inesistente ("il cavallo alato", "il cerchio quadrato"). Piú nomi possono differire per il senso e non per il significato:

"la stella della sera", "la stella del mattino", "Venere", "Espero"

sono nomi diversi dello stesso pianeta. Si accetta comunemente il

**1.1.1 Postulato** Sostituzioni *equidenotanti* all'interno di un nome non ne cambiano il significato. (Ma ne possono cambiare il senso.)

**1.1.2 Esempio** Sostituendo "il maestro di Aristotele" al posto di "Platone" nel nome (composto) "il maestro di Platone" si ottiene il nome "il maestro del maestro di Aristotele". Quest'ultimo nome differisce per il senso dal nome di partenza "il maestro di Platone", ma entrambi denotano Socrate.

**Costanti, variabili** In matematica, una *costante* è un nome provvisto di significato (dun-que non lo è "il multiplo di 5 compreso tra 6 e 9"). Una *variabile* è un nome provvisto di piú significati. La classe dei significati della variabile si chiama il suo dominio. Si assegna un valore ad una variabile quando si sceglie uno dei suoi significati. Il concetto di variabile è tanto venerabile (risale almeno ad Aristotele) ed importante (non si potrebbbe fare nessun discorso generale senza variabili) che delicato: il suo stesso nome è fonte di equivoco, giacché essa non denota affatto una quantità che varia (un numero variabile sarebbe multiplo e divisore di tutti e di nessun numero, godrebbe di tutte e di nessuna proprietà , ed un'entità cosí proteiforme sarebbe altrettanto mostruosa che inutile). In realtà la variabile va pensata come uno strumento per produrre *schemi o forme di discorso*, nei quali essa ha il ruolo di *posto* nel quale si può collocare o *sostituire* un qualunque elemento del suo dominio. Un informatico potrebbe pensare ad una variabile come ad un registro: all'assegnazione della variabile corrisponderebbe il caricamento del registro stesso.

**Forme** Sostituendo in un nome ("il padre di Pietro") ad un nome ("Pietro") una variabile $x$ che abbia nel suo dominio il nome da sostituire ("Pietro") si ottiene una forma ("il padre di $x$"). Una tale forma si dice *unaria* od anche *ad* 1 *posto*. Ripetendo l'operazione su forme 1-arie,. . . ,$n$-arie si ottengono forme binarie, . . . , $n + 1$-arie (overo, a 2,. . . , $n + 1$ posti): "il fratello di $x$ a via $y$ numero $z$". Se si assegna un sistema di valori alle variabili di una forma, questa ridiviene un nome. Il significato di quel nome è il *valore assunto*

*dalla forma* per il sistema di valori assegnato alle sue variabili. Ad esempio, se il dominio di $x$ è costituito dalle persone che hanno avuto un maestro preminente, possiamo passare dal nome "il maestro di Aristotele" alla forma "il maestro di $x$". Assegnando "Giotto" ad $x$ si ottiene il nome "il maestro di Giotto", il quale ha un suo significato (Cimabue). Si assume naturalmente di disporre di meccanismi che precludano la formazione di nomi privi di significato, sicché per esempio non si dovrebbe poter passare da "l'attuale regina d'Inghilterra" a "l'attuale regina di $x$", per finire con "l'attuale regina d'Italia".
Due forme sono *concordanti* se le loro variabili sono definite sugli stessi domini ed assumono gli stessi valori in corrispondenza degli stessi sistemi di valori per le loro variabili. Per esempio, se $h$ e $k$ sono definite su $\mathcal{N}$ sono concordanti la forma $\sum_{1\leq i\leq h} i$ e la forma $\frac{1}{2}k(k+1)$.

**1.1.3 Postulato** Sostituzioni concordanti all'interno di forme concordanti danno forme concordanti.

**Funzioni, operatore di astrazione** Una funzione $f$ è un criterio che permette di associare agli elementi di una data classe (detta *dominio* della funzione) precisamente un elemento di un'altra data classe (*codominio*). La differenza tra forma e funzione è che la prima è un fatto linguistico, dipendente entro certi limiti dal linguaggio adottato, mentre l'altra no (il nesso che c'è tra 2 e 4, 3 e 9, 4 e 16, etc. lo capiamo tutti nello stesso modo, ma può essere spiegato in vari modi). Una funzione $f$ può essere vista come una classe i cui elementi sono coppie: il primo e il secondo elemento di ciascuna coppia sono rispettivamente un elemento del dominio e del codominio di $f$. Possiamo allora estendere alle funzioni la nozione di uguaglianza estensionale data all'inizio: due funzioni sono uguali se sono insiemi formati dalle stesse coppie. La funzione associata ad una data forma è la funzione che associa, ai valori assunti dalle variabili della forma, il valore assunto dalla forma per quei valori. Se $F(x)$ è una forma, $\lambda x[F(x)]$ (o, piú brevemente, $\lambda x.F(x)$) è la funzione associata a $F(x)$. Ha allora senso scrivere

$$\lambda h. \sum_{1\leq i\leq h} i = \lambda k. \frac{1}{2}k(k+1).$$

$\lambda$ viene chiamato *operatore d'astrazione*. Con questo nome si intende forse dire che esso consente di astrarre dal particolare veicolo linguistico usato per descrivere la funzione, per cogliere un nesso strutturale tra enti matematici. Vista la nostra definizione di uguaglianza tra funzioni, e visto che due forme concordanti possono avere un senso diverso ma danno luogo alla stessa funzione associata, si può forse addirittura sostenere che si astrae dal senso.

L'uso dell'operatore di astrazione permette di risolvere questioni apparentemente oziose, come ad esempio se $1 - \sin^2 x$ e $\cos^2 y$ sono o no la stessa cosa. Sono due forme diverse, mentre le funzioni $\lambda x.1 - \sin^2 x$ e $\lambda y. \cos^2 y$ sono uguali.

**Enunciati, Tesi di Frege, valori di verità** Un *enunciato* è un'entità linguistica che si possa sensatamente giudicare vera o falsa (2+2=4, "piove" sono enunciati. 2+2 e "non rubare" non lo sono). La seguente Tesi è comoda tecnicamente.

**Tesi di Frege** Gli enunciati sono nomi.

Applicando il postulato 1.1.1 alla tesi di Frege riesce difficile assegnare significati diversi agli enunciati seguenti:
Romolo è Romolo; Romolo è il primo dei sette re di Roma; 7 è il numero dei re di Roma che cominciano con Romolo; 7 è il numero dei peccati capitali; 7=7.
La Tesi di Frege sembra allora suggerire che tutti gli enunciati sono nomi delle due entità note come il *Vero* e il *Falso*. D'ora in poi le denoteremo rispettivamente con 0 e con 1. Quando dal contesto si capisce che 0 ed 1 sono usati come *valori di verità* (e non, per esempio, come *bit*), $W$ denoterà l'insieme $\{0, 1\}$.
Gli enunciati differiscono tra loro per il senso. Si dovrebbe chiamare *proposizione* il senso di un enunciato, ma la distinzione tra enunciati e proposizioni non è molto rispettata.
(Un aspetto speculativo delle idee sin qui accennate è che per ogni *cosa*, e dunque per ogni *concetto*, ci sono infiniti nomi. Alcuni sono noti, altri ancora no: ognuno ne coglie un aspetto particolare. Con l'andare del tempo collettivo ed individuale, si aggiungono nuovi sensi, senza che le caratteristiche di alcunché possano mai essere esaurite. Questo varrebbe in particolare per la verità , la cui conoscenza parziale è continuamente accresciuta dagli enunciati veri prodotti da ciascuno di noi.)

**Variabili proposizionali, funzioni di verità** Una *variabile proposizionale* $[p, q]$ (ma si dovrebbe dire: enunciativa) è una variabile che ha per dominio $W$. Può accadere che in un enunciato figuri un altro enunciato ("piove" figura in "se piove mi bagno"). Per la Tesi di Frege, si può sostituire in un enunciato uno o piú enunciati con un numero adeguato di variabili proposizionali, ottenendo cosí una forma proposizionale: "se $p$ allora $q$". Si chiama funzione di verità ($n$-aria) una funzione associata ad una forma proposizionale (ad $n$ posti).

## 1.2 Elementi di metalogica

**Metateorie** Una qualunque teoria $\mathbf{T}$ può essere vista come una classe di espressioni (in genere, ma non sempre, enunciati) di un linguaggio $L$. Esse si riferiscono ad un universo del discorso $U$, e sono, in genere, prodotte mediante un apparato deduttivo $D$, specifico per la teoria $\mathbf{T}$. A sua volta, $D$ consiste in genere di elementi di partenza o assiomi e di regole deduttive (in senso lato). Descriviamo ciò , dicendo che $\mathbf{T}$ è una terna della forma $(L, U, D)$. Si suppone che un qualche rapporto tra gli elementi di $L$ e quelli di $U$ sia stato stabilito prima della costituzione di $\mathbf{T}$. (Un matematico, un fisico, Chiambretti usano linguaggi diversi, parlano di cose diverse e ragionano in modo diverso.)
L'universo del discorso di una teoria $\mathbf{T}$ può a sua volta essere una teoria $\mathbf{T}^o$, che si chiama allora *teoria oggetto*, mentre la teoria $\mathbf{T}$ si chiama allora la *metateoria* di $\mathbf{T}^o$; la situazione si presenta cosí :

$$\mathbf{T}^m = (L^m, (L^o, U^o, R^o), R^m),$$

dove gli indici in alto stanno rispettivamente per *metateorico* e per *oggetto.* Lo studio dell'inglese in una scuola italiana è un esempio di questa situazione. $L^o$ è l'insieme delle parole inglesi; $U^o$ è l'insieme delle cose del mondo ("dog" è in $L^o$, mentre i cani sono in $U^o$); $R^o$ è costituito dalle regole della lingua inglese. $L^m$ è l'italiano, piú qualche termine del gergo dei grammatici; etc. In vista del seguito del discorso, vale la pena di osservare che nella metateoria di una lingua si distingue una parte che si chiama *Sintassi*, che studia la forma delle parole e delle frasi, prescindendo dal loro significato (il participio si forma attaccando da qualche parte un "ing"; con "he", la negazione si fa con "doesn't", etc.); e una seconda parte, detta *Semantica* che studia il significato delle parole (nella sintassi c'è "*cani*" ma non ci sono cani).

**Sistema formale** Un *sistema formale* è una teoria priva di universo del discorso. Gli elementi del suo linguaggio sono dunque privi di un significato (almeno ufficiale). Un sistema formale è per lo piú oggetto di uno studio metateorico. Il linguag gio di un sistema formale è in genere un linguaggio formale (ossia un linguaggio generabile in modo meccanico). Sono un esempio di sistema formale gli scacchi: il linguaggio è l'insieme delle posizioni dei pezzi; le regole dell'apparato deduttivo stabiliscono la posizione iniziale e le mosse; una partita è una deduzione.
Un altro esempio è una grammatica $G = (S, T, NT, P)$; il linguaggio di un tale sistema formale (che è una cosa diversa da $L(G)$) è l'insieme delle parole in $T \cup NT$; le regole deduttive sono costituite dal *seme* o assioma $S$, e dalle *produzioni* $P$. Si deduce, con una derivazione, una parola di $L(G)$.

**Sintassi, Semantica** La metateoria di un sistema formale si chiama *sintassi*. In genere, il suo linguaggio è un linguaggio naturale ampliato, e le sue regole deduttive sono quelle di una matematica alquanto costruttiva; ossia di una matematica che non parla di enti e proprietà che non sappia in qualche modo, parzialmente o totalmente meccanico, costruire o decidere. La sua logica è un misto di rigore e di spazio per l'intuizione, paragonabile a quello comunemente adottato in algebra.
Si chiama *semantica* una metateoria di un sistema formale la quale, a differenza dalla sintassi, per prima cosa fissi un universo e definisca un'operazione di interpretazione del linguaggio del sistema formale oggetto nell'universo fissato. (Confrontando con la nozione generale di teoria di pag. 6, si vede che lí si parla di un rapporto prestabilito, almeno parzialmente, tra linguaggio e universo, mentre qui, invece, l'universo è fissato dopo che il sistema formale è stato costituito.)
In genere, la costruttività richiesta in sede semantica è piú debole che in sintassi (per esempio nella dimostrazione del teorema di completezza della Sez. **??** c'è un passaggio non costruttivo — vedi Nota **??**).

**Linguaggio metalogico** Nel seguito ci occuperemo di teorie oggetto nei cui linguaggi figurano i segni $\sim, \wedge, \vee, \rightarrow, \leftrightarrow, \forall, \exists$. Allo scopo di evitare confusioni tra il livello oggetto e il livello sintattico, non useremo questi segni con il ruolo in un certo senso *stenografico* abituale nei corsi di matematica. Conveniamo invece di adottare, come ampliamenti convenzionali significanti del linguaggio sintattico, i termini *non* , *et* , *vel* , *seq* , *aeq* ,*om* ,*ex* in luogo, rispettivamente, di quelli appena elencati.
In tutte le definizioni scriviamo *in corsivo* il nome delle entità in corso di definizione. Uno o piú simboli, racchiusi tra parentesi quadre, possono seguire: sono notazioni usate, eventualmente con indici in basso, da quel punto in poi, per denotare quelle entità, senza ulteriori indicazioni, salvo, talora, per aumentare la comprensibilità. (Vedi Es. **??**.)

# Chapter 2

# Teorie del primo ordine

## 2.1 Linguaggio

**2.1.1** I termini sono l'elemento di partenza di una qualunque teoria del primo ordine. Ne definiamo la forma generale, ed uniamo alla definizione degli esempi del modo in cui questa forma si specializzi nel linguaggio e nell'apparato deduttivo di teorie particolari, come la *teoria (formale) dei numeri*.

1. *Una variabile (soggettiva) libera* $[a, b, c, a_1, ..]$ è un elemento di una classe FV di segni.

2. Un *funtore ad* $n \geq 0$ *posti* $[f^{(n)}, g^{(n)}, f^{(n)_1}, ..]$ è un elemento di una classe $\mathrm{FNT}^{(n)}$ (vuota, finita o numerabile). I funtori a zero posti sono anche detti *costanti*. L'indice in alto $(n)$ sarà omesso quando noto o privo di interesse.

3. Un *termine*$[s, t]$ è una variabile libera o un'espressione nella forma

$$f^{(n)}t_1, \ldots, t_n \qquad (n \geq 0).$$

*Commento.* Le variabili libere e i funtori sono segni denotati da variabili sintattiche $a, g, ..$ che non verranno quasi mai esibiti. Cosí come non c'è bisogno di esibire Bari per dire che c'è una piazza davanti alla sua stazione.

**2.1.2 Esempio** Otteniamo i *termini aritmetici* ponendo

$$\mathrm{FNT}^{(0)} = \{0\}; \quad \mathrm{FNT}^{(1)} = \{'\}; \quad \mathrm{FNT}^{(2)} = \{+, \cdot\}; \quad \mathrm{FNT}^{(n)} = \emptyset \ (n > 2).$$

Per migliorare la leggibilità, adottiamo le seguenti notazioni

1. $(t)'$ sta per $'t$, mentre $(s + t)$ e $(s \cdot t)$ stanno per $+st$ e per $\cdot st$.

2. tutte le parentesi vengono omesse, salvo quelle che contrastano la priorità di $\cdot$ sul $+$ e quella di $'$ su $\cdot$.

3. $\overline{n}$ sta per $0'..'$ $\quad (n \geq 0$ occorrenze di $')$.

Un esempio di termine aritmetico è fornito dall'espressione seguente $(\overline{3} + a)' \cdot b'$ che è diverso da $((\overline{3} + a') \cdot b)'$.
Avremmo anche potuto adottare una costante per ogni numero. Questo avrebbe evitato la definizione di numerale, ma avrebbe comportato altre difficoltà , come per esempio la necessità di infiniti assiomi per dire che $f^{(0)}_{i+1}$ è il successore di $f^{(0)}_i$.

**2.1.3** 1. *Una variabile (soggettiva) vincolata* $[u, .., z, z_1, ..]$ è un elemento di una classe BV di segni.

2. Una *lettera predicativa ad n posti* $[P^{(n)}, Q^{(n)}, P^{(n)_1,..}]$ è un elemento di una classe $\mathrm{LP}^{(n)}$ (vuota, finita o numerabile). I funtori a zero posti sono anche detti *costanti*. L'indice in alto $(n)$ sarà omesso quando noto o privo di interesse.

3. Un *atomo non negato* $[\alpha, \beta, ..]$ è un'espressione della forma $P^{(n)}(t_1, \ldots, t_n)$ nella quale ogni $t_i$ è un termine, oppure è il risultato della sostituzione in un atomo non negato di una variabile libera con una variabile vincolata. Gli *atomi* sono espressioni nella forma $\alpha$ o nella forma $\neg\alpha$.

**Esempio 2.1.2 (segue)** Otteniamo gli *atomi aritmetici* ponendo $\mathrm{LP}^{(2)} = \{=\}$, e lasciando vuote tutte le altre classi $\mathrm{LP}^{(n)}$. Scrivendo $s = t$ invece di $= st$ abbiamo che $\overline{0} = \overline{1}$ e $a = b + b$ sono atomi aritmetici.

**2.1.4** 1. Una *formula* $[A, \ldots, D]$ è un atomo oppure un'espressione in una delle forme seguenti ($\oplus$ sta per $\wedge, \vee$ e $Q$ per $\exists, \forall$)

$$\sim \alpha; \quad (A \oplus B); \quad (QxA).$$

2. $\sim A$ è una notazione sintattica definita come segue

$$\begin{array}{rcl} \neg\neg B & \text{è} & B \\ \neg(B \vee C) & \text{è} & \neg B \wedge \neg C \\ \neg(B \wedge C) & \text{è} & \neg B \vee \neg C \\ \neg\exists x B & \text{è} & \forall\neg x B \\ \neg\forall x B & \text{è} & \exists\neg x B. \end{array}$$

L'espressione $A \to B$ è una notazione sintattica per la formula $\sim A \vee B$.

3. Si omettono le parentesi ridondanti rispetto alle priorità seguenti: $\sim \forall, \exists, \wedge, \vee, \rightarrow$, e rispetto alla associatività di $\wedge, \vee$. (Attenzione che invece $\rightarrow$ non è associativa.)

4. In una formula $Qx\, A$, $\quad A$ è il *raggio d'azione* del *quantificatore* $Q$ indicato.

5. La *altezza* $ht(A)$ di $A$ è 0 se $A$ è un atomo; ed è $1 + \max_i(ht(B_i))$ se $A$ è nella forma $B_0 \oplus B_1$ ovvero $QxB_0$.

6. Una formula è *aperta (chiusa)* se (non) vi figurano variabili libere.

7. **Restrizione** Non è ammessa alcuna formula con variabili vincolate $z$ che non siano nel raggio di un quantificatore $Q\ z$ (per la stessa $z$).

**2.1.5 Osservazione** Abbiamo chiesto che ogni variabile vincolata figuri nel raggio di un quantificatore, ma non che, inversamente, ogni quantificatore quantifichi realmente una variabile. Vedi Oss. 2.2.2 per le ragioni di questa asimmetria.

**Esempio 2.1.2 (segue)** Sono esempi di *formule aritmetiche* le espressioni

$$a = 0; \quad (\sim a = b \vee a = c) \wedge b = c; \quad \exists x\, x = a \wedge a = b.$$

Non sono ammesse come formule le espressioni

$$a = x; \quad \exists x\, x = a \wedge a = x; \quad \exists x(x = a \wedge a = y).$$

È ammessa $\exists x\, a = b$ (vedi Oss. 2.1.5).

## 2.2 Sostituzione

**2.2.1 Definizione** Sia $E$ un termine o una formula. Con la notazione composta $E(a, b, ..)$ si intende che le variabili $a, b, ..$ possono (non necessariamente tutte o sole) figurare in $E$. Dato $E(a, b, ..)$, si denota con $E(s, t, ..)$ il risultato della *sostituzione* in $E$ di $a, b, ..$ con $s, t, ...$ Esso è ottenuto prendendo i valori assegnati alla prima occorrenza nel discorso di $E$ e:

1. lasciando $E$ invariato se $a, b, ..$ non vi figurano;

2. altrimenti, rimpiazzando nel valore assegnato ad $E$ ogni occorrenza di $a, b, ..$ con i valori assegnati ai termini $s, t, ...$

Per esempio, se si assegna a $t(a,b)$ il valore $a+b$ si ha che $t(b+b,b)$ è $b+b+b$; ma poi, poiché si fa sempre riferimento al primo valore assegnato dal discorso alle variabili sintattiche, $t(b+b,c)$ è $b+b+c$ e non $c+c+c$.

**2.2.2 Osservazione** Definendo in questo modo la sostituzione si ha il vantaggio di avere un operatore *totale*, ossia definito per ogni argomento $A, b, t$. Al prezzo che, nel momento in cui si passa da $A(b)$ a $QxA(x)$, si deve ammettere la possibilità che venga quantificata una variabile $x$ che non figura realmente in $A$ (vedi Oss. 2.1.5). Un pò come scrivere $\sum_{1\leq i\leq n} m$, cosa che in matematica non fa scandalo.

**2.2.3** Ad ogni scelta di FNT e LP è associato un *linguaggio del primo ordine*. Per esempio:

1. Il linguaggio dell'aritmetica formale è ottenuto con i funtori $0, ', +, \cdot$ e con un'unica lettera predicativa per l'uguaglianza. (cf. 2.1.2).

2. Nel linguaggio della teoria formale degli insiemi non c'è alcun funtore. Le uniche lettere predicative sono a due posti e corrispondono all'uguaglianza e all'appartenenza. Scrivendole in forma infissa si ottengono atomi che possono essere solo nella forma $s \in t$ oppure $s = t$, dove $s$ e $t$ sono variabili libere o vincolate.

3. Nel *Calcolo dei Predicati* non ci sono funtori, ma per ogni $n$ ci sono (potenzialmente) infinite lettere predicative ad $n$ posti. I suoi termini si riducono dunque alle variabili, mentre gli atomi consistono in lettere predicative ad $n$ posti applicate ad $n$ variabili libere o vincolate (tuttavia, per la Restrizione 2.1.4.7 un atomo con una variabile vincolata $x$ non può *vivere da solo*, ma è ammesso solo come sotto-formula di una formula $QxB$).

## 2.3 Interpretazione

In questa sezione si stabilisce il modo di effettuare l'operazione semantica consistente nell'assegnare dei valori di un dominio $D$ ai termini, e dei valori di verità ($\in W = \{0,1\}$) alle formule di un linguaggio del primo ordine.

**2.3.1 Definizione** Una *interpretazione* di un linguaggio del primo ordine $L$ in un dominio $D$ è una quaterna $\Im^D = (\Im_v^D, \Im_c^D, \Im_f^D, \Im_l^D)$, dove

1. $\Im_v^D : FV \mapsto D$ è una *interpretazione delle variabili soggettive* libere;

2. $\Im_c^D : FNT^{(0)} \mapsto D$ è una *interpretazione delle costanti* .

3. $\Im_f^D$ è una *interpretazione dei funtori* ad $n > 0$ posti, formata da una classe di funzioni

$$\Im_f^{D\,(n)} : FNT^{(n)} \mapsto \{F : D^n \mapsto D\},$$

che associa ad ogni funtore ad $n$ posti una funzione da $D^n$ a $D$.

4. $\Im_l^D$ è una *interpretazione delle lettere predicative* ad $n$ posti, formata da una classe di funzioni

$$\Im_l^{D\,(n)} : LP^{(n)} \mapsto \{F : D^n \mapsto W\},$$

che associa ad ogni lettera predicativa a $n$ posti una relazione a $n$ posti definita su $D$.

Una volta scelta $\Im^D = (\Im_v^D, \Im_c^D, \Im_f^D, \Im_l^D)$, il valore $\Im^D t$ di ogni termine $t$ e il valore di verità $\Im^D A$ di ogni formula $A$ è determinato induttivamente dalle definizioni che seguono.

**2.3.2** *Interpretazione dei termini*

$$\begin{aligned} \Im^D a &= \Im_v^D a \\ \Im^D f^{(0)} &= \Im_c^D f^{(0)} \\ \Im^D f^{(n)}(t_1, \ldots, t_n) &= \Im_f^{D\,(n)} f^{(n)}(\Im^D t_1, \ldots, \Im^D t_n). \end{aligned}$$

**2.3.3** *Interpretazione delle formule senza quantificatori*

$$\begin{aligned} \Im^D P^{(n)}(t_1, \ldots, t_n) &= \Im_l^{D\,(n)} P^{(n)}(\Im^D t_1, \ldots, \Im^D t_n) \\ \Im^D \sim \alpha &= 1 - \Im^D \alpha \\ \Im^D B \vee C &= \Im^D B \cdot \Im^D C \\ \Im^D B \wedge C &= \Im^D B \dot{+} \Im^D C, \end{aligned}$$

dove la *somma logica* $a \dot{+} b$ dei bit $a$ e $b$ vale 0 se $a = b = 0$, ed 1 altrimenti.

**2.3.4 Esercizio** Per $D = \{0, 1\}$ poniamo

$$\Im_l^D P = \text{“}\textit{diverso}\text{”}; \quad \Im_v^D a = 0; \quad \Im_v^D b = 0.$$

$\Im^D P(a, b)$ è vero o falso?

**2.3.5 Definizione** Data una interpretazione $\Im^D$ ed una variabile $a$, una *reinterpretazione* è un'operazione $\Re_a$ che associa a $\Im^D$ un'altra interpretazione $\overline{\Im}^D = \Re_a \Im^D$ tale che $\Im^D$ e $\overline{\Im}^D$ sono *uguali dappertutto salvo, forse, in* $a$; ossia, in simboli

$$\Im_c^D = \overline{\Im}_c^D; \quad \Im_f^D = \overline{\Im}_f^D; \quad \Im_l^D = \overline{\Im}_l^D; \quad b \neq a \ \textit{seq}\ \Im_v^D b = \overline{\Im}_v^D b.$$

**2.3.6 Definizione** Per ogni $B(a)$ (nella quale, lo ricordiamo, non è detto che $a$ figuri)

$$\begin{cases} \Im^D \exists x B(x) = 0 & \text{se esiste } \Re_a \Im^D \text{ tale che } \Re_a \Im^D B(a) = 0 \\ \Im^D \forall x B(x) = 0 & \text{se per ogni } \Re_a \Im^D \text{ si ha } \Re_a \Im^D B(a) = 0. \end{cases}$$

**2.3.7 Esempio** Verifichiamo che per le $\Im^D$ dell'esercizio 2.3.4 si ha

$$\Im^D \forall x \exists y P(x, y) = 0.$$

A tal fine, la seconda riga di Def. 2.3.6 richiede che per ogni $\Re_a \Im^D$ si abbia $\Re_a \Im^D \exists y P(a, y) = 0$. Le $\Re_a \Im^D$ sono le interpretazioni $\Im_0^D$ e $\Im_1^D$ che assegnano ad $a$ rispettivamente 0 e 1, e sono in tutto il resto uguali ad $\Im^D$. Dobbiamo controllare che si abbia (i) $\Im_0^D \exists y P(a, y) = 0$; ed anche (ii) $\Im_1^D \exists y P(a, y) = 0$.
Per (i), la prima riga di Def. 2.3.6 richiede che esista $\Re_b \Im_0^D$ tale che $\Re_b \Im_0^D B(a, b) = 0$. Di $\Re_b \Im_0^D$ ce ne sono due, ossia $\Im_{00}^D$ e $\Im_{01}^D$ le quali assegnano 0 ad $a$ e rispettivamente 0 e 1 a $b$. Dobbiamo dunque verificare che si abbia

$$\Im_{00}^D P(a, b) = 0 \text{ oppure } \Im_{01}^D P(a, b) = 0.$$

Dall'Eserc. 2.3.4 sappiamo come verificare che la seconda uguaglianza è vera, e la prima no.

**2.3.8 Esercizio** Concludere la discussione dell'esempio precedente dimostrandone la parte (ii).

**2.3.9 Lemma** Per ogni $A$ e per ogni $\Im^D$ si ha $\Im^D \sim A = 1 - \Im^D A$.

*Dimostrazione.* Induzione su $ht(A)$. Base. Per Def. 2.3.3. Passo.
Caso 1. La forma di $A$ è $B_0 \oplus B_1$. Poniamo $a := \Im^D A$; $b_i := \Im^D B_i$. L'ipotesi induttiva (I.I.) dà $\Im^D \sim B_i = 1 - b_i$. Se $\oplus$ è $\wedge$, abbiamo

$$\begin{aligned} \Im^D \sim A &= \Im^D \sim B_0 \vee \sim B_1 && \text{(parte 2 di Def. 2.1.4 )} \\ &= (1 - b_0)(1 - b_1) && \text{(Def. 2.3.3, I.I.)} \\ &= 1 - (b_0 \dot{+} b_1) && \text{(proprietà di } \dot{+}) \\ &= 1 - a && \text{(ancora Def. 2.3.3 ).} \end{aligned}$$

Simmetricamente se $\oplus$ è $\vee$, sfruttando la proprietà $(1 - b_0) \dot{+} (1 - b_1) = 1 - b_0 b_1$.
Caso 2. Esiste $B(a)$, tale che $A$ è nella forma $\forall x B(x)$ e dunque $\sim A$ è $\exists x \sim B(x)$. Supponiamo inoltre che $\Im^D A = 1$. Per Def. 2.3.6 esiste $\overline{\Im}^D = \Re_a \Im^D$ tale che $\overline{\Im}^D B(a) = 1$. L'I.I. vale per *qualsiasi interpretazione* di qualsiasi formula $C$ tale che $ht(C) < ht(A)$.

Abbiamo dunque $\overline{\Im}^D \sim B(a) = 0$. Ancora per Def. 2.3.6 abbiamo allora $\Im^D \exists x \sim B(x) = 0$.
Simmetricamente per il sotto-caso $\Im^D A = 0$, e per il caso in cui la forma di $A$ è $\exists x B(x)$.
*Commento.* Chi non riuscisse con un*ragionamento* a vedere le proprietà di $\dot{+}$ ora impiegate potrà convincersene verificando i pochi casi possibili.

**2.3.10 Definizione**

1. Una $\Im^D$ è un *modello* nel dominio $D$ della formula $A$ se si ha $\Im^D A = 0$. Notazione: $Mod(\Im^D, A)$.

2. Un *contro-esempio* per $A$ nel dominio $D$ è una interpretazione in $D$ che non è un modello per $A$.

3. Può accadere che nella definizione di un linguaggio $L$ del primo ordine si sia tenuto presente un dominio $D$ e una collezione $S$ di funzioni e relazioni definite su $D$. In tal caso la interpretazione di $L$ nel dominio $D$ che associa ai funtori e alle lettere predicative gli elementi di $S$ si chiama *interpretazione voluta* o *standard* di $L$.

**2.3.11 Esempio** L'interpretazione standard $\Im^S$ del linguaggio del primo ordine per l'aritmetica di cui alla Def. 2.2.3 è quella nel dominio $\mathcal{N}$ che associa nel modo piú ovvio zero, successore, somma, prodotto ed uguaglianza ai segni $\overline{0}, ', +, \cdot, =$. Si ha $\Im^S \overline{n} = n$; $\Im^S \exists x(a + x' = b) =$"*minore*".

**2.3.12 Definizione** Una formula $A$ del calcolo dei predicati è *valida in un dominio $D$* se si ha $(om\ \Im^D)(Mod(\Im^D, A))$. Notazione $\models^D A$.
Una formula $A$ del calcolo dei predicati è *valida* se è valida in ogni dominio finito o numerabile. Notazione $\models A$.

**2.3.13 Osservazione** Talora si dice che i connettivi e i quantificatori sono *costanti logiche*, intendendosi con ciò piú o meno sostenere che tutti i ragionamenti impiegano questi oggetti nello stesso modo. Ne discende che una formula valida può essere considerata come una *legge logica*, in quanto non può essere falsificata da alcuna interpretazione dei simboli che vi figurano (visto che le interpretazioni non mettono in discussione le costanti logiche).
Questa tesi è molto discussa, sia in sede filosofica che in ambito tecnico (per esempio dalla *logica matematica intuizionista*). I suoi pro sono tuttavia prevalenti, e forse schiaccianti in ambito tecnico. Volendola condividere, occorre giustificare la restrizione a domini numerabili o finiti. In effetti, essa è necessaria perché vedremo che si dà il caso di formule

che, pur essendo valide in ogni dominio finito, ammettono controesempi in domini infiniti (non potremmo considerare *legge logica* una formula falsificabile per esempio in $\mathcal{N}$). Ed essa è sufficente: possiamo trascurare i domini di cardinalità superiore a quella di $\mathcal{N}$, perché vedremo che se una formula ammette un contro-esempio in un dominio infinito, allora ne ammette uno in un dominio numerabile.

## 2.4 Derivazioni

Esistono innumerevoli sistemi di logica del primo ordine, tra loro equivalenti. Tutte le teorie del primo ordine possono essere ottenute aggiungendo alla *struttura logica assiomatico-deduttiva* di uno qualunque di essi degli *assiomi propri* e/o delle *regole proprie di derivazione*. Nel nostro sistema si derivano insiemi $\Gamma$ di formule. La sua struttura assiomatico-deduttiva è precisata dalle definizioni che seguono.

**2.4.1 Notazione** Useremo lettere maiuscole greche e non latine per denotare insiemi di formule (e non piú, quindi, alfabeti). $\Gamma, \Delta$ sta per $\Gamma \bigcup \Delta$ e $\Gamma, A$ per $\Gamma, \{A\}$.
Identificheremo ogni disgiunzione $A_1 \vee .. \vee A_k$ che non sia nel campo di un altro simbolo logico con l'insieme $\Gamma = A_1, .., A_k$. Per esempio $\Gamma = A, B$ sta per $A \vee B$, ma anche per $B \vee A$ e per $A \vee B \vee A \vee A$.

**2.4.2 Definizione**

1. Gli *assiomi logici* sono nella forma $\Gamma, \alpha, \sim \alpha$.

2. Le *regole di derivazione (logiche)* sono riportate nella Fig. 1. L'applicazione di una di tali regole si chiama *inferenza*.

| regola | nome | abbreviazione |
|---|---|---|
| $\dfrac{\Gamma}{\Gamma, \Delta}$ | Introduzione di $\vee$ | $I_\vee$ |
| $\dfrac{\Gamma, B_0 \qquad \Gamma, B_1}{\Gamma, B_0 \wedge B_1}$ | Introduzione di $\wedge$ | $I_\wedge$ |
| $\dfrac{\Gamma, B(a)}{\Gamma, \forall x B(x)}$ | Introduzione di $\forall$ | $I_\forall$ |
| $\dfrac{\Gamma, B(t)}{\Gamma, \exists x B(x)}$ | Introduzione di $\exists$ | $I_\exists$ |

Fig. 1

**Restrizione** In $I_\forall$ la variabile $a$ non deve figurare in $\Gamma$.

**2.4.3 Definizione** Una *teoria del primo ordine* è una terna

$$\mathbf{T} = (L, AX, RD)$$

in cui $L$ è un linguaggio del primo ordine; in cui $AX$ è un insieme di assiomi, ottenuto aggiungendo degli *assiomi propri* a quelli di 2.4.2; in cui $RD$ è un insieme di regole, ottenuto aggiungendo delle *regole proprie* a quelle di 2.4.2.
Il *calcolo (puro) dei predicati* è la teoria del primo ordine $\mathbf{P}$ il cui linguaggio è stato già definito, ed in cui non ci sono né assiomi né regole proprie.

**2.4.4 Definizione** Sia data una teoria del primo ordine $\mathbf{T}$. Un *(albero di) derivazione* $d$ della formula $\Gamma \in L$ è un'assegnazione di formule $\Delta$ ai nodi di un albero binario, tale che

1. le foglie sono (associate ad) assiomi;
2. la radice è $\Gamma$;
3. ogni nodo che non è una foglia è la conseguenza (secondo una delle regole) del nodo, o dei due nodi, sovrastanti.

**Notazione** Scriviamo $d \vdash_{\mathbf{T}} \Gamma$ se $d$ è una derivazione di $\Gamma$. Omettiamo l'indice $\mathbf{T}$ quando noto o uguale a $\mathbf{P}$.

**2.4.5 Osservazione** Per ogni $\mathbf{T}$ si ha ovviamente $\vdash_{\mathbf{P}} \Delta$ *seq* $\Gamma \vdash_{\mathbf{T}} \Delta$.

**2.4.6 Problema** Se in $d \vdash A$ c'è un'$I_\forall$ che trasforma $B \vee C(a)$ in $B \vee \forall x C(x)$, può $a$ figurare in $A$?

**2.4.7** $\vdash A, \sim A$ (legge del *terzo escluso*).

*Dimostrazione.* Induzione su $ht(A)$. Base. Se la forma di $A$ è $(\neg)\alpha$, allora $A, \sim A$ è un assioma.
Passo. Caso 1. La forma di $A$ è $B_0 \oplus B_1$. L'ipotesi induttiva fornisce $d_i \vdash B_i, \sim B_i$. Se $\oplus$ è $\vee$ la derivazione voluta è

$$\dfrac{\dfrac{d_0\left\{\dfrac{\cdots}{B_0, \sim B_0}\right.}{B_0, \sim B_0, B_1} \qquad \dfrac{d_1\left\{\dfrac{\cdots}{B_1, \sim B_1}\right.}{B_1, \sim B_1, B_0}}{B_0, B_1, \sim B_0 \wedge \sim B_1}$$

Simmetricamente se $\oplus$ è $\wedge$ e si deve dunque ottenere $\vdash B_0 \wedge B_1, \sim B_0, \sim B_1$.
Caso 2. Esiste $B(a)$ tale che $A$ è $QxB(x)$. L'I.I. fornisce $d_0 \vdash B(a), \sim B(a)$. Se $A$ è $\exists x B(x)$, e dunque $\sim A$ è $\forall x \sim B(x)$, la derivazione voluta è

$$\dfrac{\dfrac{d_0 \left\{ \dfrac{\dots}{B(a), \sim B(a)} \right.}{\exists x B(x), \sim B(a)}}{\exists x B(x), \forall x \sim B(x)}$$

Simmetricamente se $Q$ è $\forall$ e si deve dunque ottenere $\vdash \forall x B(x), \exists x \sim B(x)$.

**2.4.8 Problema** Nella derivazione di $\exists x B(x) \vee \forall x \sim B(x)$ potevamo applicare subito la regola $I_\forall$ e successivamente la regola $I_\exists$? La risposta è che la questione è irrilevante, oppure che no, non potevamo?

**2.4.9 Problema** In quale caso si può correttamente inferire $\Gamma$ per $I_\exists$ da $\Gamma, B(a)$?

# Chapter 3

# Completezza e Indecidibilità del Calcolo dei Predicati

## 3.1 Completezza

Dimostriamo il teorema di completezza per il calcolo dei predicati mediante un algoritmo nondeterministico **C** (vedi Cap. **??** per la nozione di TM non deterministica).
**Notazioni** **C** è scritto in uno pseudo-Pascal senza tipi, in cui:
1. i delimitatori begin-end sono sostituiti da interlinee e indentazioni;
2. costrutti come **choose** $A \in \Delta$ ovvero **choose** $P$ **or** $Q$ stanno (non deterministicamente) per l'assegnazione a $A$ di un qualunque elemento di $\Delta$, e per la scelta tra l'esecuzione di $P$ o quella di $Q$.
3. $a_i$ denota la $i$-ma variabile libera. ATM è l'insieme degli atomi e degli atomi negati.
**Spiegazione dell'algoritmo** **C** cerca di trovare un controesempio sui naturali $\{1, .., q\}$ per ogni formula dell'entrata $\Gamma$, semplificandola finché non si deve arrendere perché ha trovato, su quel cammino non deterministico, un assioma (controllo deciso dalla booleana $s$). O finché non ci riesce (controllo deciso da $t$). Si assume che in $\Gamma$ non ci siano variabili libere.
Chiamiamo *fase* ogni ripetizione del ciclo esterno, e *sotto-fase* ogni chiamata della procedura *riduci*. Ogni fase assegna un nuovo valore a $\Gamma$, servendosi di $\Delta$ per scandirne gli elementi non atomici, in ordine non deterministico. $A = B \wedge C$ è sostituita da $B$ oppure da $C$. Questa *eliminazione di* $\wedge$ ($E_\wedge$) dà luogo (non deterministicamente) a due diverse computazioni. Su $A = \forall x B(x)$ si effettua una *eliminazione di* $\forall$ ($E_\forall$) che le sostituisce $B(a_q)$ ($q$ viene riassegnato con $q + 1$, onde garantire che ogni $E_\forall$ introduca una variabile che non figura in $\Gamma$). Infine la *eliminazione di* $\exists$ ($E_\exists$) usa $\Theta$ per associare ad ogni

$A = \exists x B(x)$ tutte le $B(a_i)$ che non sono già state considerate.

```
program  completezza;
s := true; t := true; q := 1; Θ := ∅;
while s and t do
   Δ := Γ − ATM;  Γ := Γ ⋂ ATM;  t := false
   while Δ ≠ ∅ do
      choose A ∈ Δ;
      riduci;
      if Γ = α, ¬α, Λ then s := false
   end
end;
if s = false then "assioma" else "no".
```

```
procedure  riduci;
case forma di A of
    B ∧ C        :  choose Γ := Γ, B
                    or Γ := Γ, C;  end  t := true
    ∀xB(x)       :  Γ := Γ, B(a_q);  t := true
                    q := q + 1;
    ∃xB(x)       :  Γ := Γ, A;
                    for i := 1 to q do
                       if B(a_i) ∉ Θ
                       then Γ := Γ, B(a_i);
                            Θ := Θ, B(a_i);
                            t := true
                    end {for};
    end {case};
```

**3.1.1 Lemma** Se $\mathbf{C}(\Gamma)$ ha un'uscita *no*, o una computazione infinita allora $\Gamma$ non è valida.

*Dimostrazione.* Denotiamo con $\Gamma(i), \Delta(i)$ e $\Theta(i)$ il valore di $\Gamma, \Delta$ e $\Theta$ all'inizio della fase $i$. Consideriamo la seguente interpretazione sui naturali

$$
\begin{array}{rcl}
\Im(a_j) & = & j \\
\Im(P_j^{(n)})(i_1,..,i_n) & = & 1 \quad \text{se } P_j(a_{i_1},..,a_{i_n}) \text{ è in qualche } \Gamma(i) \\
\Im(P_j^{(n)})(i_1,..,i_n) & = & 0 \quad \text{altrimenti.}
\end{array}
$$

Diciamo che la formula $A$ è *vera (falsa)* se abbiamo $\Im A = 0$ ($\Im A = 1$). Faccio vedere che ogni $A$ in ogni $\Gamma(i)$ è falsa. Scegliamo una formula $B$ di minima altezza tra quelle che (per assurdo) sono vere, e sia $B \in \Gamma(r)$. Casi secondo la forma di $B$.

Caso 1. $B$ è l'atomo non negato $\alpha$. Contraddizione perché gli atomi in $\Gamma(r)$ sono falsi.
Caso 2. La forma di $B$ è $\neg\alpha$. Siccome **C** lascia inalterati gli atomi, se avessimo $\alpha \in \Gamma(p)$ per qualche $\Gamma(p)$ avremmo $\alpha, \neg\alpha \in \Gamma(\max(r,p))$. In tal caso **C** darebbe l'uscita *assioma*. Pertanto $\alpha$ è vero e $B$ è falsa.
Caso 3. La forma di $B$ è $D_0 \wedge D_1$. **C** mette $D_i$ in $\Gamma(r+p)$ per un certo $p$ e per $i=0$ o per $i=1$. $D_i$ è falsa (perché $ht(D_i) < ht(B)$) e quindi $B$ è falsa.
Caso 4. La forma di $B$ è $\forall x D(x)$. **C** mette $D(a_i)$ in $\Gamma(r+p)$ per certi $i$ e $p$. $D(a_i)$ è falsa (perché $ht(D(a_i)) < ht(B)$) e quindi pure $B$ è falsa.
Caso 5. La forma di $B$ è $\exists x D(x)$. Sottocaso 5.1. **C** si è fermato, dopo aver assegnato a $q$ un certo valore $p$, ed avendo messo messo ogni $D(a_i)$ in qualche $\Gamma(s)$ $(i \leq p)$. Ogni $D(a_i)$ è falsa (perché $ht(D(a_i)) < ht(B)$) e quindi pure $B$ è falsa.
Sottocaso 5.2. La computazione non termina. Come per 5.1, ma osservando che ora $q$ assume ogni valore su $\mathbf{N}$.

**3.1.2 Lemma** Se $\mathbf{C}(\Gamma) =$ *assioma* per ogni computazione, allora $\vdash \Gamma$.

*Dimostrazione.* Associamo alle computazioni non deterministiche di **C** per l'entrata $\Gamma$ un albero nel modo che segue.
(1) $\Gamma$ è (assegnato a) la radice.
(2) Se *riduci* trasforma il nodo $\Gamma, \Delta, \forall x A$ in $\Gamma, \Delta, A(a_q)$ allora questo insieme è associato al padre.
(3) Se *riduci* trasforma il nodo $\Gamma, \Delta, \exists x A$ in $\Gamma, \Delta, A(a_{i_1}), .., A(a_{i_k})$ allora questo insieme è associato al padre.
(4) Se *riduci* compie una $E_\wedge$ non deterministica che trasforma il nodo $\Gamma, \Delta, B_1 \wedge B_2$ in $\Gamma, \Delta, B_i$, allora questi due insiemi sono associati ai due genitori di questo nodo binario.
Facciamo vedere che ogni nodo di TR(**C**) è derivabile. Induzione sull'altezza di questo albero. Base. Le foglie sono assiomi. Passo. Casi (2)-(4) come nella costruzione dell'albero, con le stesse notazioni.
(2) Per l'ipotesi induttiva abbiamo $\vdash \Gamma, \Delta, A(a_q)$. Poiché la costruzione di **C** garantisce che $a_q$ non figura in $\Gamma, \Delta$, possiamo ottenere $\Gamma, \Delta, \forall x A$ per $I_\forall$.
(3) Per l'ipotesi induttiva abbiamo $\vdash \Gamma, \Delta, A(a_{i_1}), .., A(a_{i_k})$. Otteniamo $\Gamma, \Delta, \exists x A$ applicando $I_\exists$ per $k$ volte.
(4) Per l'ipotesi induttiva abbiamo $\vdash \Gamma, \Delta, B_0$, nonché $\vdash \Gamma, \Delta, B_1$. Per $I_\wedge$ otteniamo $\vdash \Gamma, \Delta, B_0 \wedge B_1$.

**3.1.3 Teorema di Completezza** Esiste un algoritmo che per entrata un qualunque insieme di formule $\Gamma$ in cui non figurino variabili libere

1. se $\Gamma$ è valido si arresta fornendo le informazioni necessarie per costruire una derivazione di $\Gamma$;
2. altrimenti diverge o si arresta fornendo un contro esempio per $\Gamma$

*Dimostrazione.* Immediatamente dai due lemmi precedenti.

## 3.2 Indecidibilità

**Notazione 3.2.1** Una volta introdotta una lettera predicativa ad $n$ posti ($n$-lp) $P$, ogni occorrenza di $P$ dove ci si aspetterebbe di trovare una formula è un'abbreviazione per $P(a_1, \ldots, a_n)$. I simboli $M, N, T, a$ (magari con indici) sono variabili soggettive generiche, che mi riprometto di interpretare come stringhe che rappresentano TM, nastri e bit (talora osservati).

**Teorema 3.2.2** Il calcolo dei predicati è indecidibile. (Ma, per il teorema di completezza, è semidecidibile.)

*Dimostrazione.* Ad absurdum, per riduzione di un linguaggio indecidibile. A tal fine scriviamo alcune formule che *forzano* (enforce), mediante la parte a destra di $=_{df}$, certe $n$-lp ad esprimere alcune proprietà delle TM e dei nastri.

1. Uguaglianza. Scelta una 2-lp $E$, scriviamo $x = y$ invece di $E(x, y)$. Poniamo

   $$\mathtt{EQ} =_{df} \forall x \forall y \forall z \quad x = x \wedge (x = y \to y = x) \wedge (x = y \wedge y = z \to x = z).$$

2. Alfabeto. Date le 1-lp $L_\varepsilon$ e $L_a$ (una per ciascuna lettera $a \in \Gamma = \{A, \ldots, Z, 0, 1, \#, [, ]\}$) scriviamo $x = \varepsilon$ nonché $x = a$ invece di $L_\varepsilon(x)$ e di $L_a(x)$. Scriviamo $\mathtt{Let}(x)$ per $\bigvee_{a \in \Gamma} x = a$. Poniamo

   $$\mathtt{GAMMA} =_{df} \forall x (\mathtt{Let}(x) \wedge \bigwedge_{a,b \in \Gamma; a \neq b} x = a \leftrightarrow a \neg b).$$

3. Concatenazione. Scegliamo una 3-lp $C$, e scriviamo $x = yz$ per $C(x, y, z)$. Poniamo

   $$\begin{aligned} \mathtt{CONC} \quad =_{df} \quad \forall x \forall y \forall z \quad & x = y = z = \varepsilon \vee (y = \varepsilon \wedge x = z) \vee (z = \varepsilon \wedge x = y) \\ & \vee \exists u \exists v \exists w (\mathtt{Let}(u) \wedge y = uv \wedge x = uw \wedge w = vz). \end{aligned}$$

   Notazioni per la concatenazione di $n > 2$ stringhe

   $$\begin{array}{rll} x = x_1 \ldots x_{n+1} & \text{sta per} & \exists u (u = x_1 \ldots x_n \wedge x = u x^{n+1}) \\ y = x_1 \ldots x_n \wedge \ldots & \text{sta per} & \exists x_1 \ldots \exists x_n (y = x_1 \ldots x_n \wedge \ldots). \end{array}$$

4. Nastro e Macchina (nella terza riga $[N_1\ N_2]$ va intesa come *if* 0 *then* $N_1$ *else* $N_2$ mentre l'altro caso è la composizione; nella quarta $[N]$ è una *while*)

$$\begin{array}{rll} \texttt{Nstr}(T) & \leftrightarrow_{df} & T = \varepsilon \vee T = aT_1 \wedge \texttt{Nstr}(T_1) \wedge (a = 0 \vee a = 1)). \\ \texttt{Mac}(M) & \leftrightarrow_{df} & M = \varepsilon \vee M = D \vee M = S \vee M = W \vee M = E \\ & & \vee(M = N_1N_2 \vee M = [N_1N_2]) \wedge \texttt{Mac}(N_1) \wedge \texttt{Mac}(N_2)) \\ & & \vee M = [N] \wedge \texttt{Mac}(N). \end{array}$$

5. Configurazione istantanea (la TM è infissa nella configurazione alla sinistra del carattere osservato)

$$\texttt{Conf}(x) \leftrightarrow_{df} x = T_1MaT_2 \wedge \texttt{Nstr}(T_1) \wedge \texttt{Nstr}(T_2) \wedge \texttt{Bit}(a) \wedge \texttt{Mac}(M).$$

6. Next (da $x$ a $y$ in un passo)

$$\begin{array}{rll} \texttt{Next}(x,y) & \leftrightarrow_{df} & \texttt{Conf}(x) \wedge \texttt{Conf}(y) \\ & & \wedge(\texttt{Dx}(x,y) \vee \texttt{Sx}(x,y) \vee \texttt{Wrt}(x,y) \vee \texttt{Ers}(x,y) \vee \\ & & \texttt{If}(x,y) \vee \texttt{While}(x,y)) \\ \texttt{Dx}(x,y) & \leftrightarrow_{df} & = x = T_1DMabT_2 \wedge \wedge y = T_1aMbT_2 \\ \texttt{Sx}(x,y) & \leftrightarrow_{df} & = x = T_1bSMaT_2 \wedge \wedge y = T_1MbaT_2 \\ \texttt{Wrt}(x,y) & \leftrightarrow_{df} & = x = T_1WMaT_2 \wedge \wedge y = T_1M1T_2 \\ \texttt{Ers}(x,y) & \leftrightarrow_{df} & = x = T_1EMaT_2 \wedge \wedge y = T_1M0T_2 \\ \texttt{if}(x,y) & \leftrightarrow_{df} & = \wedge_a(x = T_1[N_0N_1]MaT_2 \wedge y = T_1N_aMaT_2) \\ \texttt{while}(x,y) & \leftrightarrow_{df} & x = T_1[N]M1T_2 \wedge y = T_1N[N]M1T_2 \vee \\ & & x = T_1[N]M0T_2 \wedge y = T_1M0T_2. \end{array}$$

7. Computazione accettante (# separa le configurazioni di una computazione; $M$ accetta cancellando il nastro e fermandosi sul secondo zero)

$$\begin{array}{rll} \texttt{COMPACC} & =_{df} & \forall C(\texttt{CompAcc}(C) \leftrightarrow (\texttt{Conf}(C) \wedge C = 0\varepsilon 00 \vee \\ & & C = x\#yz \wedge \texttt{Next}(x,y) \wedge \texttt{CompAcc}(yz))). \end{array}$$

8. Accettazione di 0 ($M$ accetta l'input alla sinistra dello zero che osserva)

$$\texttt{Acc}_0(M) \leftrightarrow_{df} \exists C(\texttt{CmpAcc}(C) \wedge \exists y(x = 0M00\#y)).$$

Poniamo ora $A =_{df} \texttt{EQ} \land \texttt{GAMMA} \land \texttt{CONC} \land \texttt{COMPACC}$ ed osserviamo che la formula

$$B =_{df} A \rightarrow \texttt{Acc}_0(M)$$

è valida se e solo se $M$ accetta 0. Per convincerci di questo, consideriamo un modello $\Im$ di $A$. Se $M$ accetta 0, allora $\Im$ soddisfa $\texttt{Acc}_0(M)$ ed è un modello per $B$. Se $\Im$ non è un modello per $A$ allora lo è per $B$ (perché non soddisfa l'antecedente). Dunque se $M$ accetta, $B$ è valida. Se $M$ non accetta, ogni modello di $A$ (e ce ne sono) non è un modello per $B$ che quindi non è valida.
Ne concludiamo che, decidendo la validità delle formule del calcolo dei predicati, possiamo decidere la classe $\{M \mid M(0) = 0\}$, contro il teorema di Rice.

# Chapter 4

# Incompletezza

**Premessa 4.0.3** In questo capitolo ci proponiamo di esporre dei risultati di tipo essenzialmente negativo o *limitativo*, i quali valgono per qualunque teoria del primo ordine **T** che soddisfi certe Condizioni C1,C2,D1,D2,D3 che andremo esponendo nel corso del suo svolgimento. Le prime due di esse sono sufficenti per un teorema importante, noto come *Lemma di Diagonalizzazione* e per il Teorema di Tarski. Delle altre tre, note come *condizioni di Hilbert o di derivabilità*, la prima (con $C1, C2$) è sufficente per il Primo teorema di Gödel. Infine, tutte e cinque le condizioni danno il Secondo.
Per non perdere in generalità avremmo bisogno di notazioni che valgano per linguaggi qualsiasi e tipi di dati anch'essi qualsiasi. D'altra parte, il Lettore ha già familiarità con alcune notazioni per la teoria dei numeri, e questa teoria soddisfa le cinque condizioni. **PA** può dunque svolgere il compito di *esempio di riferimento* per le considerazioni di questo capitolo. Per non moltiplicare le notazioni, adotteremo l'espediente di usare *con sistematica ambiguità* gli stessi simboli per denotare enti della particolare teoria **PA** ed enti della generica teoria **T**. È dunque richiesto al Lettore di considerare i simboli che ora introduciamo sia nel senso generale di **T**, sia nel senso particolare di **PA**, cosí come egli fa quando, parlando di strutture algebriche, accetta di vedere il segno $\times$ come moltiplicazione tra naturali o matrici, e contemporaneamente anche come segno per un'operazione astratta che soddisfa certe proprietà di associatività , etc.

## 4.1 Espressione e rappresentazione

Siano dati un dominio $D$, una teoria **T**, ed una iniezione $\lambda h.\overline{h}$ degli elementi $h$ di $D$ nei termini di **T**. Per esempio, nel caso $D = \mathcal{N}$, una tale iniezione può essere quella che manda i numeri nei numerali definiti precedentemente.

**Definizione 4.1.1** Sia data una interpretazione $\Im^D$ in $D$ del linguaggio di **T**. La formula ad $n+1$ posti $A(a_1, \dots, a_n, b)$ *esprime* la funzione ad $n$ posti $F$ se si ha

$$F(h_1, \dots, h_n) = k \ aeq \ Mod(\Im^D, A(\overline{h_1}, \dots, \overline{h_n}, \overline{k})).$$

La formula $A(a_1, \dots, a_n)$ *esprime* la relazione ad $n$ posti $\mathcal{R}$ se si ha

$$\begin{array}{rll} \mathcal{R}(h_1, \dots, h_n) & aeq & Mod(\Im^D, A(\overline{h_1}, \dots, \overline{h_n})) \\ non\ \mathcal{R}(h_1, \dots, h_n) & aeq & Mod(\Im^D, \sim A(\overline{h_1}, \dots, \overline{h_n})). \end{array}$$

Nel resto di questo capitolo, diremo che una formula è *vera* se è soddisfatta dalla interpretazione della quale stiamo parlando.

**Esempio 4.1.2** Nella interpretazione standard di **PA**, la formula $\overline{i} \cdot \overline{j} \cdot a \cdot a \cdot b \cdot b \cdot b = c$ esprime il polinomio $\lambda hk.ih^2jk^3$, e la formula $a \prec b$ esprime la relazione $h < k$.
Non è immediato stabilire se la funzione $\lambda hk.h^k$ è esprimibile o no in **PA**. (La risposta è sí.)

*Commento.* Questa definizione prescinde completamente dall'apparato deduttivo di **T**: essa stabilisce un rapporto tra il suo linguaggio ed il nostro. La prossima definizione invece prescinde da ogni interpretazione, e si limita al piano della discussione sintattica dell'apparato deduttivo.

**Definizione 4.1.3** La formula $A(a_1, \dots, a_n, b)$ *rappresenta* in **T** la funzione ad $n$ posti $F$ se

$$\begin{array}{rll} F(h_1, \dots, h_n) = k & seq & \vdash_{\mathbf{T}} A(\overline{h_1}, \dots, \overline{h_n}, \overline{k}) \\ F(h_1, \dots, h_n) \neq k & seq & \vdash_{\mathbf{T}} \sim A(\overline{h_1}, \dots, \overline{h_n}, \overline{k}). \end{array}$$

$A(a_1, \dots, a_n)$ *rappresenta* la relazione ad $n$ posti $\mathcal{R}$ se si ha

$$\begin{array}{rll} \mathcal{R}(h_1, \dots, h_n) & seq & \vdash_{\mathbf{T}} A(\overline{h_1}, \dots, \overline{h_n}) \\ non\ \mathcal{R}(h_1, \dots, h_n) & seq & \vdash_{\mathbf{T}} \sim A(\overline{h_1}, \dots, \overline{h_n}) \end{array}$$

**Osservazione 4.1.4** Per ogni sistema di valori per $h_1, \dots, h_n$ ed eventualmente $k$ è permessa una derivazione diversa. Il Lemma **??** dimostra che $a + b$ rappresenta in **PA** la somma aritmetica, mentre il Lemma **??** dimostra che $a \prec b$ vi rappresenta la relazione $h < k$.
Estendendo alla moltiplicazione il Lemma **??**, si dimostra che ogni polinomio in $n$ variabili $p(h_1, \dots, h_n)$ è espresso e rappresentato in **PA** da un atomo (vedi anche Es. 4.1.2) della forma

$$t(a_1, \dots, a_n) = b.$$

Non è vero l'inverso, nel senso che per esempio la funzione caratteristica del *minore*

$$c_<(h,k) := \mathit{if}\ h < k\ \mathit{then}\ 0\ \mathit{else}\ 1$$

è rappresentata dalla formula non atomica

$$(\exists x(a_1 + x' = a_2) \rightarrow b = 0) \wedge (\sim \exists x(a_1 + x' = a_2) \rightarrow b = 1).$$

Onde la notazione seguente, la quale consente di trattare come termini oggetti che, a rigore, tali non sono.

**Notazione 4.1.5** Se $B(a_1, \ldots, a_n, b)$ esprime o rappresenta in $\mathbf{T}$ la funzione $F$, allora

| | | |
|---|---|---|
| $A(\overline{F}(a_1, \ldots, a_n))$ | denota la formula | $\exists x(A(x) \wedge B(a_1, \ldots, a_n, x))$ |
| $A(\overline{F(h_1, \ldots, h_n)})$ | denota la formula | $\exists x(A(x) \wedge B(\overline{h_1}, \ldots, \overline{h_n}, x))$. |

Questo ci consente di dire che la funzione $F(h_1, \ldots, h_n)$ è espressa o rappresentata dallo *pseudo-termine* $\overline{F(h_1, \ldots, h_n)}$ (invece di dire dalla formula stabilita nelle Def. 4.1.1 e 4.1.3). Introduciamo ora la prima delle Condizioni per i teoremi limitativi. Il suo senso è che linguaggio ed apparato deduttivo di $\mathbf{T}$ devono saper *identificare* le funzioni (che sanno esprimere o rappresentare) con i loro valori (cfr. Oss. **??**, facendo attenzione che nella formulazione della prossima Cond. 1, la quantificazione "*per ogni* $A(a)$ *è metamatematica*").

Scriviamo $A \leftrightarrow B$ come abbreviazione per $A \rightarrow B \wedge B \rightarrow A$.
**Condizione** C1 Se $\overline{F}$ esprime o rappresenta $F$ allora per ogni $A(b)$, si ha (rispettivamente) la seguente *Condizione semantica,* o, rispettivamente, *sintattica di mutua sostituibilità tra pseudo-termini e loro valori*

| | | | | |
|---|---|---|---|---|
| C1.1 | $F(h_1, \ldots, h_n) = k$ | *seq* | $A(\overline{F(h_1, \ldots, h_n)}) \leftrightarrow A(\overline{k})$ | è vera; |
| C1.2 | $F(h_1, \ldots, h_n) = k$ | *seq* | $A(\overline{F(h_1, \ldots, h_n)}) \leftrightarrow A(\overline{k})$ | è derivabile in $\mathbf{T}$. |

## 4.2 Diagonalizzazione

**Definizione 4.2.1** Una *codifica* o *gödelizzazione* di una teoria $\mathbf{T}$ in un dominio $D$ è una iniezione $\lambda E.\ulcorner E \urcorner$ delle formule di $\mathbf{T}$ e delle loro componenti sintattiche (variabili, funtori, etc.) negli elementi di $D$.

**Notazione 4.2.2** 1. $A_n$ è la formula codificata da $n$, sicché si ha $\ulcorner A_n \urcorner = n$.

2. $A[a_1, \dots, a_k]$ ($k \geq 0$) significa che le $a_1, \dots, a_k$ sono tutte e sole le variabili libere di $A(a_1, \dots, a_k)$.

**Definizione 4.2.3** Dati $\mathbf{T}$ e una sua gödelizzazione in un dominio $D$, si chiama *sostituzione diagonale* una funzione $sb$ tale che

$$\begin{cases} \text{se } A_n \text{ è una formula } B[a_1, \dots, a_k], \text{ allora} & sb(n) := \ulcorner B(\overline{n}, \dots, \overline{n}) \urcorner \ (k \text{ volte}) \\ \text{altrimenti} & sb(n) := \ulcorner \epsilon \urcorner. \end{cases}$$

**Condizione** C2 Sono dati una Gödelizzazione di $\mathbf{T}$ in $D$ ed uno pseudo-termine $\overline{sb}$ che

C2.1 *(condizione semantica di diagonalizzabilità)* esprime in $\mathbf{T}$ la funzione $sb$.

C2.2 *(condizione sintattica di diagonalizzabilità)* rappresenta in $\mathbf{T}$ la funzione $sb$.

**Teorema 4.2.4** (*Lemma di diagonalizzazione*) Ogni formula può essere resa auto-referenziale. Ossia, per ogni formula $A[b]$ esiste una formula $B$ chiusa (cfr. Def. 2.1.4.6) tale che

1. se in $\mathbf{T}$ valgono le condizioni 1.1 e 2.1, allora $A(\overline{\ulcorner B \urcorner}) \leftrightarrow B$ è vera.

2. se in $\mathbf{T}$ valgono le condizioni 1.2 e 2.2 allora $\vdash_{\mathbf{T}} A(\overline{\ulcorner B \urcorner}) \leftrightarrow B$.

*Dimostrazione.* Costruzione di $B$. Definiamo la *diagonalizzazione* $A^\delta$ di $A$ con

$$A^\delta(a) := A(\overline{sb}(a)). \tag{4.1}$$

Poniamo ora

$$m := \ulcorner A^\delta \urcorner; \qquad B := A^\delta(\overline{m}).$$

Osservando che $B$ è ottenuta sostituendo $\overline{m}$ in $A_m(a)$, vediamo dalla Def. 4.2.3 che si ha

$$sb(m) = \ulcorner B \urcorner. \tag{4.2}$$

La formula $B$ è chiusa perché l'unica variabile libera di $A$ è stata assegnata con $\overline{m}$. Verifichiamo ora che essa gode delle proprietà asserite (ognuna delle righe che seguono afferma che la formula che vi figura è *derivabile* ovvero *vera*, secondo la condizione in ipotesi)

| | | |
|---|---|---|
| 1 | $B \leftrightarrow A^\delta(\overline{m})$ | $B$ e $A^\delta(\overline{m})$ sono due diverse notazioni per la stessa formula |
| 2 | $B \leftrightarrow A(\overline{sb(m)})$ | (4.1), transitività di $\leftrightarrow$ |
| 3 | $B \leftrightarrow A(\overline{\ulcorner B \urcorner})$ | (4.2), Condizione 1, transitività di $\leftrightarrow$. |

*Commento.* Il risultato è sorprendente. Esso dice che se $A[a]$ decide una proprietà $\mathcal{P}$ allora c'è una formula $B$ che equivale all'enunciato $\mathcal{P}$ *è vera del codice di* $B$. Si ponga

attenzione tuttavia che esso non dice né che $B$ è vera, né che è derivabile: solo $A(\overline{\ulcorner B \urcorner}) \leftrightarrow B$ lo è.
Se per esempio $A(a)$ è $\forall y \forall z(y \langle a \wedge z \langle a \rightarrow \sim a = y \cdot z)$, e quindi esprime la proprietà *h è un numero primo*, allora il teorema afferma che $B$ è vera se e solo se il suo codice è un numero primo (e non che è vero che il suo codice è un numero primo). Ed effettivamente nella gödelizzazione piú comune di **PA**, *nessuna* formula è codificata da un numero primo, e quindi sia $\forall y \forall z(y \langle \overline{\ulcorner B \urcorner} \wedge z \langle \overline{\ulcorner B \urcorner} \rightarrow \sim \overline{\ulcorner B \urcorner} = y \cdot z)$ che $B$ sono false.

## 4.3 Le definizioni di verità non sono esprimibili

**Definizione 4.3.1** Date una teoria $\mathbf{T}$ ed una sua interpretazione $\Im$, la proprietà
$\mathcal{P}(n) := Mod(\Im, A_n)$ *et* $A_n$ è una formula chiusa
è una *definizione di verità* (per $\mathbf{T}$ relativa a $\Im$).

Sia data una una teoria $\mathbf{T}$ ed una sua interpretazione $\Im$ che soddisfino le versioni semantiche delle Condizioni 1 e 2.

**Teorema 4.3.2** *(di Tarski) La definizione di verità relativa a* $\Im$ *non è esprimibile in* $\mathbf{T}$.

*Dimostrazione.* La formula $Tr(a)$ esprima (ad abs.) in $\mathbf{T}$ una definizione di verità, sicché

$$Tr(\overline{n}) \quad \text{è vera se e solo se} \quad A_n \quad \text{è vera.} \tag{4.3}$$

Esiste, per il teorema di diagonalizzazione, una formula chiusa $T?$ tale che

$$T? \leftrightarrow \sim Tr(\overline{\ulcorner T? \urcorner}) \quad \text{è vera.} \tag{4.4}$$

Ma una contraddizione nasce dall'ipotesi che $T?$ sia vera (perché allora $Tr(\overline{\ulcorner T? \urcorner})$ è vera per la (4.3) e falsa per la (4.4)). Una simmetrica contraddizione discende anche dall'ipotesi che $T?$ sia falsa. L'asserto segue perché $T?$ è chiusa, e non ci sono dunque altre possibilità.
*Commento.* Si intenda il carattere corsivo come un modo per dare un nome (codificare) enti ed in particolare enunciati del linguaggio naturale. Ha senso affermare che

$$\textit{piove} \text{ è un enunciato vero se e solo se piove.} \tag{4.5}$$

Si potrebbe sostenere che i paradossi e le oscurità che si rilevano nelle scienze deduttive siano attribuibili alla scarsa definitezza del linguaggio naturale; si potrebbe allora preconizzare un programma di riduzione di tali scienze ad una o piú teorie espresse in linguaggi del primo ordine, auspicatamente *perfetti*. Se ci fosse una definizione di verità, i loro contenuti coinciderebbero con la classe delle formule vere. Il teorema dice che non è possibile adattare ai linguaggi del primo ordine definizioni che seguano l'idea implicita nella (4.5). Prima ancora di cominciare a discutere l'apparato deduttivo di una teoria $\mathbf{T}$, dobbiamo prendere atto che c'è già qualcosa che non può neanche essere *detto* in $\mathbf{T}$.

## 4.4 Incompletezza

**Definizione 4.4.1** Una teoria **T** è *(sintatticamente) completa* se ogni sua formula chiusa vi è formalmente decidibile, ossia vi è derivabile o vi è confutabile.
**T** è *coerente* se nessuna formula $A$ vi è derivabile e confutabile.

*Commento.* Il requisito della completezza per una teoria del primo ordine è diverso da quello per il calcolo puro dei predicati, perché in **P**:
(1) si chiede la verità in tutte le $\Im$ (validità );
(2) essendoci delle formule chiuse che non sono né valide né contraddittorie (non soddisfacibili), non si può avere la completezza sintattica.
Osserviamo infine che la coerenza è banalmente ottenibile con un apparato deduttivo troppo povero; la completezza lo è con un apparato troppo ricco. Si pone il problema di trovare un equilibrio.

Sia data una teoria **T** che soddisfa le varianti sintattiche delle Condizioni 1 e 2.
**Condizione D1** Definiamo la seguente relazione tra elementi di $D$

$$prov(h,k) := h = ^{\ulcorner} d^{\urcorner} \ et \ d \vdash A_k.$$

La relazione *prov* è rappresentata in **T** dallo pseudo-termine $\overline{prov}$, sicché

$$\begin{array}{rcl} d \vdash A & seq & \vdash_{\mathbf{T}} \overline{prov(\ulcorner d \urcorner, \ulcorner A \urcorner)} \\ non\ d \vdash A & seq & \vdash_{\mathbf{T}} \sim \overline{prov(\ulcorner d \urcorner, \ulcorner A \urcorner)} \end{array} \tag{4.6}$$

Poniamo

$$pr(a) := \exists x \overline{prov}(x,a). \tag{4.7}$$

$D1 \qquad \vdash A \ seq \ \vdash pr(\overline{\ulcorner A \urcorner})$.
Nel resto di questo capitolo ci limitiamo ad interpretazioni tali che $\Im(\overline{prov}) = prov$.

**Definizione 4.4.2** Una teoria è *ricca* se, per ogni sua tesi di carattere esistenziale $\exists x A(x)$, c'è un'altra tesi $A(\overline{n})$ che fornisca un codice come esempio di tale tesi esistenziale. Ossia se

$$(om\ A(b))(\vdash \exists x A(x) \ seq \ (ex\ \overline{n})(\vdash A(\overline{n}))).$$

**Esempio 4.4.3** Supponiamo che **PA** sia ricca, e supponiamo che vi si riesca ad ottenere $\vdash pr(\overline{\ulcorner 0 = 1 \urcorner})$. Poiché *pr* è una formula a carattere esistenziale, ci deve essere un *capro espiatorio* per questa sciocchezza, ossia un $\overline{n}$ tale che $\vdash \overline{prov(n, \ulcorner 0 = 1 \urcorner)}$.

**T** soddisfi anche $D1$ (oltre quanto già detto per le le Condizioni 1 e 2).

**Teorema 4.4.4** *(di Incompletezza, anche Primo teorema di Gödel)*

1. *Se* **T** *è coerente, esiste una formula chiusa e vera* $G$ *che non vi è derivabile.*

2. *Se* **T** *è ricca, allora* $G$ *non vi è nemmeno confutabile.*

*Dimostrazione.* Sia $G$ la formula chiusa fornita dal lemma di diagonalizzazione, tale che

$$\vdash G \leftrightarrow \sim pr(\overline{\ulcorner G \urcorner}). \tag{4.8}$$

1 Si abbia (ad abs.) $\vdash G$. Abbiamo

| | | |
|---|---|---|
| 1 | $\vdash pr(\overline{\ulcorner G \urcorner})$ | D1 |
| 2 | $\vdash \sim G$ | (4.8),1,logica |

contraddizione con l'ipotesi che **T** è coerente.
2 Si abbia (ad abs.) $\vdash \sim G$. Abbiamo

| | | |
|---|---|---|
| 1 | $\vdash pr(\overline{\ulcorner G \urcorner})$ | ipotesi $\vdash \sim G$, (4.8), logica |
| 2 | $(ex\ n) \vdash \overline{prov(n, \ulcorner G \urcorner)}$ | 1, Def. 4.4.2, si veda anche Es. 4.4.3 |
| 3 | $\vdash \sim \overline{prov(n, \ulcorner G \urcorner)}$ | parte 1 del teorema, seconda riga di (4.6) |

ancora una contraddizione con l'ipotesi che **T** è coerente.

## 4.5 Non derivabilità della coerenza

**Condizione D2** *Formalizzazione del modus ponens* Per tutte le formule $A, B$

$$\vdash pr(\overline{\ulcorner A \urcorner}) \wedge pr(\overline{\ulcorner A \to B \urcorner}) \to pr(\overline{\ulcorner B \urcorner}.$$

**Condizione D3** *Hull condition* [1] Per ogni formula $A$

$$(omA) \vdash pr(\overline{\ulcorner A \urcorner}) \to pr(\overline{\ulcorner pr(\overline{\ulcorner A \urcorner}) \urcorner}).$$

Sia data una teoria **T** che soddisfi le cinque condizioni precedenti. Poniamo

$$\perp := \overline{\ulcorner G \wedge \sim G \urcorner}; \qquad Con := \sim pr(\perp). \tag{4.9}$$

La formula $Con$ è vera se e solo se **T** è coerente.

[1] *Hull* = guscio in inglese.

**Teorema 4.5.1** *(Secondo teorema di Gödel, anche Corollario di Gödel)*

$$\textit{Se } \mathbf{T} \textit{ è coerente allora} \quad \textit{non } \vdash_{\mathbf{T}} Con.$$

*Dimostrazione.* Sia (ad abs.) $\vdash Con$. Il lavoro principale da compiere è far vedere che

$$\vdash Con \to G. \tag{4.10}$$

Infatti (4.10), unita a $\vdash Con$, fornisce $\vdash G$, in contraddizione col Teor. 4.4.4.
L'dea intuitiva della derivazione di $Con \to G$ è che essa, in un certo senso, *formalizza* la prima metà del Teor. 4.4.4, giacché *dice* che "*se* $\mathbf{T}$ *è coerente allora* $G$", mentre $G$ *dice* che "$G$ *non è derivabile*" (si veda anche il Commento dopo il Teor. 4.2.4). Ma se una formula $A$ formalizza un teorema, ci si può attendere che la formalizzazione della dimostrazione di quel teorema fornisca una derivazione di $A$. Dividiamo in tre parti la dimostrazione di (4.10).

| | | | |
|---|---|---|---|
| (1) | 1 | $\vdash pr(\overline{\ulcorner G\urcorner}) \to\sim G$ | (4.8), logica |
| | 2 | $\vdash pr(\overline{\ulcorner pr(\overline{\ulcorner G\urcorner}) \to\sim G)\urcorner})$ | 1,D1 |
| | 3 | $\vdash pr(\overline{\ulcorner G\urcorner}) \to pr(\overline{\ulcorner pr(\overline{\ulcorner G\urcorner})\urcorner})$ | D3 |
| | 4 | $\vdash pr(\overline{\ulcorner pr(\overline{\ulcorner G\urcorner})\urcorner}) \wedge pr(\overline{\ulcorner pr(\overline{\ulcorner G\urcorner}) \to\sim G\urcorner}) \to pr(\overline{\ulcorner\sim G\urcorner})$ | D2 |
| | 5 | $\vdash pr(\overline{\ulcorner G\urcorner}) \to pr(\overline{\ulcorner\sim G\urcorner})$ | 3,4,CUT, ancora CUT con 2 |
| (2) | 6 | $\vdash G \to (\sim G \to G\wedge \sim G)$ | logica |
| | 7 | $\vdash pr(\overline{\ulcorner G \to (\sim G \to G\wedge \sim G)\urcorner})$ | 6,D1 |
| | 8 | $\vdash pr(\overline{\ulcorner G\urcorner}) \to pr(\overline{\ulcorner\sim G \to G\wedge \sim G\urcorner})$ | 7,D2,logica |
| | 9 | $\vdash pr(\overline{\ulcorner G\urcorner}) \to (\overline{\ulcorner\sim G\urcorner}) \to pr(\perp)$ | 8,D2,logica |
| (3) | 10 | $\vdash pr(\overline{\ulcorner G\urcorner}) \to pr(\perp)$ | 5,9,CUT,consolidamento |
| | 11 | $\vdash Con \to\sim pr(\overline{\ulcorner G\urcorner})$ | 10,(4.9),logica |

A questo punto, la riga 11 fornisce la (4.10) per logica e (4.8).