

Logica Matematica

Sapienza Università di Roma

libro del corso: tbd

aglaia norza

10 febbraio 2026

Indice

1 Logica Proposizionale	3
1.1 Introduzione	3
1.2 Assegnamenti, tavole di verità	4
1.3 Conseguenza logica	5
1.4 Completezza funzionale	7
1.5 Forme normali	8
1.6 Equivalenza Logica	10
1.7 Formalizzazioni in logica proposizionale	10
1.7.1 Esempi di formalizzazioni di problemi noti in Logica Proposizionale	11
1.8 Teorema di compattezza	14
1.8.1 Dimostrazione per i linguaggi numerabili	15
1.8.2 Dimostrazione per i linguaggi arbitrari	16
1.9 Applicazioni del teorema di compattezza	19
1.9.1 Lemma di König	21
1.10 Decidibilità	23
2 Calcolo Proposizionale	25
2.1 Teorema di completezza	26
2.1.1 Teorema di Deduzione	27
2.1.2 Completezza Semplice	28
2.1.3 Conclusione del Teorema di Completezza	29
3 Logica Predicativa	31
3.1 Introduzione	31
3.2 Semantica	33

3.3	Teorie	34
3.4	Isomorfismi tra strutture	36
3.5	La teoria DLO	40
3.5.1	Numeri razionali	40
3.6	DLO: caso generale	41
3.6.1	Sottostrutture, proprietà del testimone	42
3.6.2	Equivalenza tra \mathcal{R} e \mathcal{Q}	44
3.6.3	Generalizzazione della dimostrazione	45
3.7	Criterio di Tarski-Vaught	46
3.8	Teorema di Löwenheim-Skolem (All'in giù)	47
3.9	Teorie categoriche (ω -categoricità)	48
4	Calcolo dei Predicati	50
4.0.1	Proprietà fondamentali del Calcolo dei Predicati	51
4.0.2	Teorema di completezza per la logica predicativa	51
4.0.3	Estensioni di teorie	52
4.1	Teorema di compattezza	57
4.1.1	Applicazione: (non) assiomatizzabilità	58
4.1.2	Modelli non standard dell'aritmetica	59
5	I teoremi di Gödel	61
5.1	Funzioni calcolabili (algoritmiche)	61
5.2	Teorema di Definibilità	62
5.3	I numeri di Gödel	64
5.4	Indecidibilità algoritmica dell'aritmetica (Tarski)	65
5.4.1	Prima forma debole di Incompletezza	66
5.5	Primo Teorema di Incompletezza di Gödel	67
5.5.1	Aritmetica minimale (MA)	67
5.5.2	Il teorema	68

1.1. Introduzione

La logica proposizionale è un linguaggio formale con una semplice struttura sintattica basata su proposizioni elementari (atomiche) e sui seguenti connettivi logici:

- *Negazione* (\neg): inverte il valore di verità di un enunciato: se un enunciato è vero, la sua negazione è falsa, e viceversa.
- *Congiunzione* (\wedge): il risultato è vero se e solo se entrambi i componenti sono veri.
- *Disgiunzione* (\vee): il risultato è vero se almeno uno dei componenti è vero.
- *Implicazione* (\rightarrow): rappresenta l'enunciato logico “se ... allora”. Il risultato è falso solo se il primo componente è vero e il secondo è falso.
- *Equivalenza* (\leftrightarrow): rappresenta l'enunciato logico “se e solo se”. Il risultato è vero quando entrambi i componenti hanno lo stesso valore di verità, cioè sono entrambi veri o entrambi falsi.

Introduciamo anche il concetto di disgiunzione esclusiva o “XOR” (\oplus), il cui risultato è vero solo se gli operandi sono diversi tra di loro (uno vero e uno falso).

Def. 1: Linguaggio proposizionale

Un linguaggio proposizionale è un insieme infinito \mathcal{L} di simboli detti **variabili proposizionali**, tipicamente denotato come $\{p_i : i \in I\}$ (con I “insieme di indici”).

Def. 2: Proposizione

Una **proposizione** in un linguaggio proposizionale è un elemento dell’insieme PROP così definito:

- (1) tutte le variabili appartengono a PROP
- (2) se $A \in \text{PROP}$, allora $\neg A \in \text{PROP}$
- (3) se $A, B \in \text{PROP}$, allora $(A \wedge B), (A \vee B), (A \rightarrow B) \in \text{PROP}$
- (4) nient’altro appartiene a PROP (PROP è il più piccolo insieme che contiene le variabili e soddisfa le proprietà di chiusura sui connettivi 1 e 2)

Per facilitare la leggibilità delle formule, definiamo le seguenti regole di *precedenza*: \neg ha precedenza su \wedge, \vee , e questi ultimi hanno precedenza su \rightarrow .

1.2. Assegnamenti, tavole di verità

Per un linguaggio \mathcal{L} , un **assegnamento** è una funzione

$$\alpha : \mathcal{L} \rightarrow \{0, 1\}$$

Estendiamo α ad $\hat{\alpha} : \text{PROP} \rightarrow \{0, 1\}$ in questo modo:

- $\hat{\alpha}(\neg A) = \begin{cases} 1 & A = 0 \\ 0 & A = 1 \end{cases}$
- $\hat{\alpha}(A \wedge B) = \begin{cases} 1 & \hat{\alpha}(A) = \hat{\alpha}(B) = 1 \\ 0 & \text{altrimenti} \end{cases}$
- $\hat{\alpha}(A \vee B) = \begin{cases} 0 & \hat{\alpha}(A) = \hat{\alpha}(B) = 0 \\ 1 & \text{altrimenti} \end{cases}$
- $\hat{\alpha}(A \rightarrow B) = \begin{cases} 0 & \hat{\alpha}(A) = 1 \wedge \hat{\alpha}(B) = 0 \\ 1 & \text{altrimenti} \end{cases}$

notazione

Utilizzeremo α al posto di $\hat{\alpha}$ per comodità di notazione.

Osserviamo che è possibile rappresentare gli assegnamenti in modo compatto utilizzando le **tavole di verità**, una presentazione tabulare della funzione di assegnamento.

Per esempio, possiamo riscrivere la definizione di $\alpha(\neg A)$ come segue:

A	$\neg A$
0	1
1	0

Ogni riga di una tavola di verità corrisponde ad un assegnamento α .

Si noti anche che dalla definizione di α segue che un'implicazione può essere vera senza che ci sia connessione causale o di significato tra antecedente e conseguente (per esempio, “se tutti i quadrati sono pari allora π è irrazionale”).

In secondo luogo, segue anche che una proposizione è sempre vera se il suo antecedente è falso (il che rispecchia la pratica matematica di considerare vera a vuoto una proposizione ipotetica la cui premessa non si applica).

Questo è giustificabile come segue:

- vogliamo che $(A \wedge B) \rightarrow B$ sia sempre vera
- il caso $1 \rightarrow 1$ deve essere vero, perché corrisponde al caso in cui A e B sono vere;
- il caso $0 \rightarrow 0$ deve essere vero, perché corrisponde al caso in cui $A \wedge B$ è falso perché B è falso; il caso $0 \rightarrow 1$ deve essere vero perché corrisponde al caso in cui $A \wedge B$ è falso perché B è falso;

il caso $0 \rightarrow 1$ deve essere vero perché corrisponde al caso in cui $A \wedge B$ è falso perché A è falso ma B è vero;

resta dunque soltanto il caso $1 \rightarrow 0$, che non corrisponde a nessun caso di $A \wedge B \rightarrow B$.

In più, si vuole che valga, per contrapposizione $(A \rightarrow B) \rightarrow (\neg B \rightarrow \neg A)$.

Osserviamo che, data $A = p_1, p_2, \dots, p_k$ e due assegnamenti α e β t.c.:

$$\begin{aligned}\alpha(p_1) &= \beta(p_1) \\ &\dots \\ \alpha(p_k) &= \beta(p_k)\end{aligned}$$

allora necessariamente $\alpha(A) = \beta(A)$.

soddisfacibilità

Se per una formula A e un assegnamento α si ha $\alpha(A) = 1$, si dice che “ A soddisfa α ” (o “ A è vera sotto α ”).

- Se A ha almeno un assegnamento che la soddisfa, si dice **soddisfacibile** ($A \in \text{SAT}$).
- Se non esiste un assegnamento che la soddisfa, A si dice **insoddisfacibile** ($A \in \text{UNSAT}$).
- Se A è soddisfatta da tutti i possibili assegnamenti, si dice **tautologia** (o “verità logica”) ($A \in \text{TAUT}$).

Introduciamo anche alcune regole che

1.3. Conseguenza logica

Def. 3: Conseguenza logica

Sia T una *teoria*, ossia un insieme $\{A_1, \dots, A_n\}$ proposizioni in un dato linguaggio proposizionale, e sia $A \in \text{PROP}$.

Diciamo che A è **conseguenza logica** di T se

$$\forall \alpha, \alpha(T) = 1 \rightarrow \alpha(A) = 1$$

ovvero se ogni assegnamento che soddisfa T soddisfa anche A_{n+1} .

Scriviamo in tal caso $T \models A_{n+1}$, oppure $A_1, \dots, A_n \models A$.

Si ha che:

- $T \not\models A$ significa che $\exists \alpha$ t.c. $\alpha(T) = 1 \wedge \alpha(A) = 0$
- $\emptyset \models A$ o, equivalentemente $\models A \iff A$ è una tautologia
- se $T \models A$, allora $T \cup \neg A$ è insoddisfacibile
- la conseguenza logica ha la proprietà di **monotonia**: se $T \models A$, allora anche $T \cup B \models A$
- ha anche la proprietà di transitività: se $T \models A$ e $A \models B$ allora $T \models B$

Lemma 1: Equivalenze

- (1) $T \vDash A$
- (2) $\vDash (A_1 \wedge \dots \wedge A_n) \rightarrow A$
- (3) $(A_1 \wedge \dots \wedge A_n \wedge \neg A) \in \text{UNSAT}$

sono equivalenti.

Operazioni sulle teorie (utili per esercizi)

Vediamo come si comportano le teorie in base a operatori noti:

Unione di teorie ($T \cup S$):

L'unione rappresenta una sorta di “aggiunta di vincoli”: si sta chiedendo che valgano sia le proposizioni di T che quelle di S .

- monotonia - se una formula è conseguenza logica di una teoria, lo è anche dell'unione

$$T \vDash A \implies T \cup S \vDash A$$

- modus ponens - se una teoria fornisce la premessa ed un'altra l'implicazione, la loro unione deriva la conclusione

$$T \vDash A \wedge S \vDash A \rightarrow B \implies S \cup T \vDash B$$

- “restringimento” dei modelli - i modelli dell'unione sono l'intersezione dei modelli

$$\text{Mod}(T \cup S) = \text{Mod}(T) \cap \text{Mod}(S)$$

- “incoerenza” - se sia T che S sono coerenti, non è detto che la loro unione lo sia

esempio: $T = \{p\}$, $S = \{\neg p\}$

Intersezione di teorie ($T \cap S$):

L'intersezione tra teorie è “debole”, in quanto contiene solo le proposizioni che appartengono ad entrambi gli insiemi.

- “perdita di informazioni” - l'intersezione tra teorie è sempre un sottoinsieme delle teorie originali (banale)

$$T \cap S \subseteq T, S$$

- coerenza - se T è coerente, allora $T \cap S$ è necessariamente coerente (non stiamo aggiungendo informazioni)

Esempio:

premesse	derivano dalle premesse?	sì/no	motivo
$T \vDash A$	$T \cup S \vDash A$	V	Monotonia
$T \vDash A$	$T \cap S \vDash A$	F	Serve che anche $S \vDash A$
$T \vDash A \wedge S \vDash \neg A$	$T \cup S$ è insoddisfacibile	V	Contraddizione ($A \wedge \neg A$)
$T \vDash A \wedge S \vDash \neg A$	$T \cap S \vDash A \leftrightarrow \neg A$	F	$T \cap S$ è coerente
$A \text{ sat} \iff B \text{ sat}$	$T \cap S \vDash (A \leftrightarrow B)$	F	$\text{Sat} \neq \text{Equivalenza}$
$T, A \vee B \vDash C \wedge S \vDash B$	$T \cup S \vDash A \rightarrow C$	F	A potrebbe non implicare C

1.4. Completezza funzionale

Data una tavola di verità arbitraria con n argomenti, esiste una proposizione A che ha esattamente quella tavola di verità?

Una proposizione A contenente le n variabili proposizionali a_1, a_2, \dots, a_n determina una funzione di n argomenti $f : \{0, 1\}^n \rightarrow \{0, 1\}$ (“**funzione di verità**”), tale che il valore di f_A su un argomento $(x_1, x_2, \dots, x_n) \in \{0, 1\}^n$ sia dato da un arbitrario assegnamento α tale che $\alpha(p_k) = x_k$ per $k \in [1, n]$.

Thm. 1: Teorema

Sia $f : \{0, 1\}^n \rightarrow \{0, 1\}$ una funzione di verità. Esiste una proposizione A con n variabili proposizionali tale che, per ogni assegnamento α :

$$\alpha(A) = f(\alpha(a_1), \alpha(a_2), \dots, \alpha(a_n))$$

(per qualsiasi funzione di verità, esiste sempre una proposizione che si comporta esattamente come essa (il valore di verità di A , $\alpha(A)$ è uguale al risultato della funzione f a parità di input))

dimostrazione

Si dimostra per induzione su n .

- **caso base:** $n = 1$ abbiamo quattro possibili f :

$$\begin{array}{ll} f_1(0) = 0, & f_1(1) = 0 \\ f_2(0) = 1, & f_2(1) = 1 \\ f_3(0) = 0, & f_3(1) = 1 \\ f_4(0) = 1, & f_4(1) = 0 \end{array}$$

Alla funzione f_1 corrisponde la formula $(p \wedge \neg p)$, alla funzione f_2 la formula $(p \vee \neg p)$, alla funzione f_3 la formula p , e alla funzione f_4 la formula $(\neg p)$.

- **caso induttivo:** (assumiamo che il teorema valga per $n - 1$ variabili, e dimostriamo che vale per n)

Se $n > 1$, scriviamo il grafico di

$$f : \{0, 1\}^n \rightarrow \{0, 1\}$$

in forma di tavola di verità in questo modo:

p_1	p_2	\cdots	p_n	$f(p_1, \dots, p_n)$	
0	0	...	
⋮		⋮		⋮	grafico di una funzione f_0
0	1	...	
1	0	...	
⋮		⋮		⋮	grafico di una funzione f_1
1	1	...	

Se non consideriamo la prima colonna (p_1), la tavola di verità descrive il grafico di due funzioni, f_0 e f_1 , a $n - 1$ argomenti.

Sappiamo, quindi, per ipotesi induttiva, che esistono due formule A_0 e A_1 a $n - 1$ variabili tali che, per ogni assegnamento α :

$$\alpha(A_0) = f_0(\alpha(p_1), \alpha(p_2), \dots, \alpha(p_n))$$

$$\alpha(A_1) = f_1(\alpha(p_1), \alpha(p_2), \dots, \alpha(p_n))$$

Dobbiamo ora combinare le due formule considerando anche la colonna p_1 .

Possiamo farlo tramite la formula $A = (\neg p_1 \rightarrow A_0) \wedge (p_1 \rightarrow A_1)$.

Dimostriamo che A soddisfa il teorema: dobbiamo dimostrare che, dato un assegnamento qualsiasi α , si ha:

$$\alpha(A) = f(\alpha(p_1), \alpha(p_2), \dots, \alpha(p_n))$$

Distinguiamo i due casi:

- $\alpha(p_1) = 1$

in questo caso, si ha:

$$\alpha\left(\underset{=1}{\neg p_1} \rightarrow A_0 \wedge \underset{=1}{p_1} \rightarrow A_1\right)$$

e la formula vale quindi $1 \iff \alpha(A_1) = 1$.

Ma $\alpha(A_1) = f_1(\alpha(p_2), \dots, \alpha(p_n))$, quindi la formula si comporta esattamente come f_1 :

$$f(\alpha(p_1), \alpha(p_2), \dots, \alpha(p_n)) = f(1, \alpha(p_2), \dots, \alpha(p_n)) = f_1(\alpha(p_2), \dots, \alpha(p_n)).$$

Quindi, in questo caso, vale

$$\alpha(A) = (\alpha(p_1), \alpha(p_2), \dots, \alpha(p_n))$$

- $\alpha(p_1) = 0$

in questo caso, si ha:

$$\alpha\left(\underset{=1}{\neg p_1} \rightarrow A_0 \wedge \underset{=1}{p_1} \rightarrow A_1\right)$$

che vale $1 \iff \alpha(A_0) = 1$.

Quindi si può fare lo stesso ragionamento di sopra, ma per A_0 e f_0 .

Potremmo anche costruire una funzione f che rappresenta il comportamento di A :

$$f(x_1, x_2, \dots, x_n) = \begin{cases} f_1(x_2, \dots, x_n) & \text{se } x_1 = 1, \\ f_0(x_2, \dots, x_n) & \text{se } x_1 = 0. \end{cases}$$

1.5. Forme normali

notazione

Chiamiamo “letterale” una variabile proposizionale o una negazione di una variabile proposizionale

È utile individuare alcune forme normali canoniche.

Def. 4: Forma Normale Disgiuntiva

Diciamo che A è in Forma Normale Disgiuntiva (**DNF**, *Disjunctive Normal Form*) se A è una disunzione di congiunzioni di letterali, ossia è nella forma seguente:

$$\bigvee_{i \leq n} \bigwedge_{j \leq m_i} A_{ij} = (A_{1,1} \wedge \cdots \wedge A_{1,m_1}) \vee \cdots \vee (A_{n,1} \wedge \cdots \wedge A_{n,m_n})$$

Def. 5: Forma Normale Congiuntiva

Diciamo che A è in Forma Normale Congiuntiva (**CNF**, *Conjunctive Normal Form*) se A è una disunzione di congiunzioni di letterali, ossia è nella forma seguente:

$$\bigwedge_{i \leq n} \bigvee_{j \leq m_i} A_{ij} = (A_{1,1} \vee \cdots \vee A_{1,m_1}) \wedge \cdots \wedge (A_{n,1} \vee \cdots \vee A_{n,m_n})$$

1.6. Equivalenza Logica

Def. 6: Equivalenza logica

Due formule $A, B \in \text{PROP}$ sono logicamente equivalenti ($A \equiv B$) quando, per ogni assegnamento α si ha $\alpha(A) = \alpha(B)$.

Introduciamo alcune regole utili per verificare l'equivalenza tra proposizioni.

Con un piccolo abuso di notazione, definiamo 1 e 0 come le formule per cui $\forall \alpha, \alpha(1) = 1$ e $\alpha(0) = 0$.

In questo modo, abbiamo:

Involuzione	$\neg\neg A \equiv A$
Assorbimento (con 0 e 1)	$A \vee 0 \equiv A$ $A \wedge 1 \equiv A$
Cancellazione	$A \vee 1 \equiv 1$ $A \wedge 0 \equiv 0$
Terzo escluso (tertium non datur)	$A \vee \neg A \equiv 1$ $A \wedge \neg A \equiv 0$
Leggi di De Morgan	$\neg(A \vee B) \equiv \neg A \wedge \neg B$ $\neg(A \wedge B) \equiv \neg A \vee \neg B$
Commutatività	$A \vee B \equiv B \vee A$ $A \wedge B \equiv B \wedge A$
Associatività	$A \vee (B \vee C) \equiv (A \vee B) \vee C$ $A \wedge (B \wedge C) \equiv (A \wedge B) \wedge C$
Distributività	$A \vee (B \wedge C) \equiv (A \vee B) \wedge (A \vee C)$ $A \wedge (B \vee C) \equiv (A \wedge B) \vee (A \wedge C)$
I teorema di assorbimento	$A \vee (A \wedge B) \equiv A$ $A \wedge (A \vee B) \equiv A$
II teorema di assorbimento	$A \vee (\neg A \wedge B) \equiv A \vee B$ $A \wedge (\neg A \vee B) \equiv A \wedge B$

Tabella 1.1: Principali leggi di equivalenza logica

Nota Bene: soddisfabilità ed equivalenza logica non sono interscambiabili - una frase del tipo “Se A è soddisfacibile se e solo se B è soddisfacibile, allora $A \leftrightarrow B$ ” non è vera in assoluto (se entrambe possono essere vere non significa che lo siano nello stesso momento)

1.7. Formalizzazioni in logica proposizionale

Il concetto di soddisfabilità ci permette di usare insiemi di formule proposizionali per catturare determinate strutture matematiche.

Per esempio: sia X un insieme. Consideriamo il linguaggio proposizionale composto dalle variabili $p_{(x,y)}$ per ogni $(x, y) \in X \times X$, e consideriamo il seguente insieme T di proposizioni in questo linguaggio:

- (1) $\neg p_{x,x} \forall x \in X$ (antiriflessività)
- (2) $p_{x,y} \rightarrow \neg p_{y,x} \forall x \in X$ (asimmetria)
- (3) $(p_{x,y} \wedge p_{y,z}) \rightarrow p_{x,z} \forall x, y, z \in X$ (transitività)
- (4) $(p_{x,y} \vee p_{y,x}) \forall x \neq y \in X$ (ordine totale)

Usiamo una teoria T per poter gestire anche casi di insiemi infiniti. Infatti, sappiamo che una teoria infinita è soddisfatta se e solo se lo sono tutte le sue proposizioni.

L'insieme $T = T_X$ esprime il concetto di **ordine totale stretto** su X . Infatti, se avessimo un assegnamento α che soddisfa tutte le proposizioni di T , l'ordine indotto da tutte le variabili vere sotto α sarebbe un ordine totale stretto di X .

Se α è un assegnamento, definiamo la relazione \prec_α su X come segue:

$$x \prec_\alpha y \leftrightarrow \alpha(p_{x,y}) = 1$$

Si ha che per ogni assegnamento α che soddisfa T_X , l'ordine \prec_α indotto da α è un ordine totale stretto su X .

Dall'altra parte, se \prec è un ordine totale stretto su X , e α_\prec è l'assegnamento indotto da \prec così definito:

$$\alpha_\prec(p_{x,y}) = 1 \leftrightarrow (x \prec y)$$

Si ha che, per ogni ordine totale stretto \prec su X , l'assegnamento α_\prec indotto da \prec sulle variabili $p_{x,y}$ soddisfa T .

Ovvero, un assegnamento α soddisfa la teoria T_X se e solo se l'ordine indotto da α su X è un ordine totale.

1.7.1. Esempi di formalizzazioni di problemi noti in Logica Proposizionale

Colorabilità

2-colorabilità

Consideriamo il problema di colorare una mappa (grafo) con due colori (Rosso e Blu) in modo che nazioni confinanti abbiano colori diversi.

Per esempio:

- mappa M : Italia (I), Austria (A), Ungheria (U).
- variabili proposizionali: X_C indica che la nazione X ha il colore C .

Per modellare il problema in logica proposizionale, definiamo tre tipi di formule:

- (1) *Ogni nazione ha almeno un colore:*

$$(I_R \vee I_B) \wedge (A_R \vee A_B) \wedge (U_R \vee U_B)$$

- (2) *Ogni nazione ha al più un colore (unicità):*

$$(I_R \rightarrow \neg I_B) \wedge (A_R \rightarrow \neg A_B) \wedge (U_R \rightarrow \neg U_B)$$

- (3) *Nazioni confinanti hanno colori diversi:*

$$(I_R \rightarrow \neg A_B) \wedge (I_B \rightarrow \neg A_R) \wedge (A_R \rightarrow \neg U_B) \wedge (A_B \rightarrow \neg U_R)$$

Esiste una corrispondenza biunivoca tra una colorazione valida e un assegnamento di verità che soddisfa la teoria:

- Una colorazione valida f induce un assegnamento α che soddisfa le formule (es. se I è rossa, $\alpha(I_R) = 1, \alpha(I_B) = 0$).
- Un assegnamento α che soddisfa le formule permette di ricostruire una colorazione valida.

La mappa M è quindi *2-colorabile* se e soltanto se l'insieme delle proposizioni è soddisfacibile.

La formalizzazione completa non è però la più “economica”. Il vincolo di unicità del colore (punto 2 sopra) è conseguenza logica degli altri vincoli (punto 1 e punto 3).

$$\text{Vincolo (1)} \wedge \text{Vincolo (3)} \models \text{Vincolo (2)}$$

È quindi possibile omettere il secondo gruppo di formule senza alterare la soddisfacibilità.

Considerando Slovenia (S), Austria (A) e Ungheria (U):

- Queste tre nazioni sono tutte confinanti tra loro (formano un triangolo nel grafo).
- L'insieme delle proposizioni risulta *insoddisfacibile*.

Infatti, se S è Rossa ($S_R = 1$), allora i vicini devono essere non-rossi ($A_R = 0, U_R = 0$). Poiché devono avere un colore, allora devono essere Blu ($A_B = 1, U_B = 1$). Ma A e U confinano, quindi non possono essere entrambi Blu. Contraddizione.

Formalizzazione della colorabilità di grafi

Sia $G = (V, E)$ un grafo e k il numero di colori disponibili ($1, \dots, k$). Definiamo le variabili $P_{v,i}$ che significano “il vertice v ha colore i ”.

L'insieme T di formule che formalizza il problema è:

- (1) **Almeno un colore per ogni vertice:**

$$\bigvee_{i=1}^k P_{v,i} \quad \text{per ogni } v \in V$$

- (2) **Al più un colore per ogni vertice (mutua esclusione):**

$$\neg(P_{v,i} \wedge P_{v,j}) \quad \text{per ogni } v \in V, \text{ con } i \neq j$$

- (3) **Vincolo di adiacenza (colori diversi per archi):**

$$\neg(P_{v,i} \wedge P_{w,i}) \quad \text{per ogni arco } \{v, w\} \in E, \text{ per ogni colore } i$$

Teorema: Il grafo G ammette una k -colorazione se e solo se l'insieme di formule T è soddisfacibile. La colorazione è data da $c(v) = i \iff \alpha(P_{v,i}) = 1$.

Pigeonhole Principle

Il Principio dei Cassetti (Pigeonhole Principle, $PHP(m, n)$) afferma che se inseriamo m oggetti (piccioni) in n cassetti, con $m > n$, allora almeno un cassetto deve contenere più di un oggetto.

Consideriamo il caso con m piccioni e n cassetti. Definiamo le variabili proposizionali:

$$p_{i,j} \quad \text{con } i \in \{1, \dots, m\} \text{ e } j \in \{1, \dots, n\}$$

Il significato intuitivo di $p_{i,j}$ è: “l’oggetto i è nel cassetto j ”.

Per formalizzare il problema, costruiamo due tipi di proposizioni:

(1) Ogni oggetto è in almeno un cassetto (Totalità)

Per un singolo oggetto i , la disgiunzione di tutti i possibili cassetti deve essere vera:

$$(p_{i,1} \vee p_{i,2} \vee \dots \vee p_{i,n})$$

Per tutti gli m oggetti, prendiamo la congiunzione:

$$A = \bigwedge_{i=1}^m \left(\bigvee_{j=1}^n p_{i,j} \right)$$

(2) Ogni cassetto contiene al più un oggetto (Iniettività)

Questa condizione esprime che non ci sono collisioni. Se questa condizione è vera, la funzione è iniettiva. Per ogni cassetto k , e per ogni coppia di oggetti distinti i e j , non è possibile che entrambi siano in k :

$$\neg(p_{i,k} \wedge p_{j,k}) \quad \text{equivalente a} \quad (\neg p_{i,k} \vee \neg p_{j,k})$$

Formalizziamo l’iniettività su tutto il dominio come la congiunzione di questi vincoli:

$$B = \bigwedge_{k=1}^n \bigwedge_{1 \leq i < j \leq m} (\neg p_{i,k} \vee \neg p_{j,k})$$

(3) Il teorema

Il principio afferma che se ogni oggetto è assegnato a un cassetto (formula A), allora l’assegnamento *non* può essere iniettivo (non B), dato che $m > n$. La formula che esprime il $PHP(m, n)$ è quindi:

$$A \rightarrow \neg B$$

(tautologia)

Spesso, nel contesto SAT, si cerca di dimostrare il principio per assurdo, cercando un assegnamento che renda vera la sua negazione. Cerchiamo cioè una situazione in cui tutti i piccioni sono assegnati (A) e l’assegnamento è iniettivo (B):

$$A \wedge B$$

In forma estesa:

$$\left(\bigwedge_{i=1}^m \bigvee_{j=1}^n p_{i,j} \right) \wedge \left(\bigwedge_{k=1}^n \bigwedge_{i \neq j} (\neg p_{i,k} \vee \neg p_{j,k}) \right)$$

Se $m > n$, questo insieme di formule è una insoddisfacibile, poiché non esiste alcun modo di mettere m piccioni in n cassetti senza collisioni.

1.8. Teorema di compattezza

Def. 7: Monotonia della conseguenza logica

Si dice che la nozione di conseguenza logica è **monotona**, ovvero che

$$T' \models A \wedge T' \subseteq T \implies T \models A$$

(se $A_1, A_2, \dots, A_k \models A$, allora $T \models A$ per ogni teoria T contenente A_1, A_2, \dots, A_k)

Nonostante non sembri intuitivamente vero, vale anche il viceversa:

Thm. 2: Teorema di compattezza v.1

Se $T \models A$, esiste un sottoinsieme finito T_0 di T tale che $T_0 \models A$

Introduciamo il concetto di una teoria finitamente soddisfacibile:

Def. 8: FINSAT

Una teoria si dice **finitamente soddisfacibile** ($\in \text{FINSAT}$) se *ogni* suo sottoinsieme finito è soddisfacibile.

Possiamo quindi introdurre una nuova versione del teorema di compattezza:

Thm. 3: Teorema di compattezza v.2

$\text{FINSAT} \implies \text{SAT}$, ovvero se ogni sottoinsieme di T è soddisfacibile, anche T è soddisfacibile.

Lemma 2: Teorema di compattezza v.1 \equiv v.2

I due punti seguenti (le due versioni del teorema di compattezza) sono equivalenti:

- (1) $T \models A \iff \exists T_0 \stackrel{\text{fin}}{\subseteq} T \text{ t.c. } T_0 \models A$
- (2) $T \in \text{SAT} \iff T \in \text{FINSAT}$

- ① \implies ②

Supponiamo per assurdo che $T \models A \implies \exists T_0 \stackrel{\text{fin}}{\subseteq} T \text{ t.c. } T_0 \models A$, che $T \in \text{FINSAT}$, ma che $T \notin \text{SAT}$ ($T \in \text{UNSAT}$).

Se $T \in \text{UNSAT}$, possiamo dire che $T \models p \wedge \neg p$ (tutto è conseguenza logica di una teoria insoddisfacibile).

Per ①, quindi, $\exists T_0 \text{ t.c. } T_0 \stackrel{\text{fin}}{\subseteq} T \models p \wedge \neg p$, il che va in contraddizione con $T \in \text{FINSAT}$.

- ② \implies ①

Supponiamo per assurdo che $T \in \text{FINSAT} \implies T \in \text{SAT}$, e che $T \models A$ ma che $\forall T_0 \stackrel{\text{fin}}{\subseteq} T, T_0 \not\models A$. $T_0 \not\models A$ significa $T_0 \cup \{\neg A\} \in \text{SAT}$.

Preso un qualsiasi sottoinsieme finito $S \subseteq (T \cup \{\neg A\})$, questo sarà contenuto in un qualche $T_0 \cup \{\neg A\}$. Dato che ogni $T_0 \cup \{A\} \in \text{SAT}$, anche $S \in \text{SAT}$. Dunque, $T \cup \{\neg A\} \in \text{FINSAT}$.

Quindi, visto che $\text{FINSAT} \implies \text{SAT}$, $T \cup \{\neg A\} \in \text{SAT}$, il che va in contraddizione con l'ipotesi $T \models A$. \square

Thm. 4: Estendibilità di SAT

Se T è soddisfacibile, allora $T \cup \{A\}$ è soddisfacibile oppure $T \cup \{\neg A\}$ è soddisfacibile.

dimostrazione dalle dispense

Sia α un assegnamento che soddisfa T . Se $\alpha(A) = 1$ allora $T \cup \{A\}$ è soddisfacibile. Se $\alpha(A) = 0$, $T \cup \{\neg A\}$ è soddisfacibile.

dimostrazione vista in classe

Supponiamo $T \in \text{SAT}$, $T \cup \{A\} \in \text{UNSAT}$ e $T \cup \{\neg A\} \in \text{UNSAT}$. Avremmo entrambi $T \models \{\neg A\}$ e $T \models A$, il che è impossibile se $T \in \text{SAT}$.

Un concetto analogo vale per FINSAT.

Thm. 5: Estendibilità di FINSAT

Sia $T \in \text{FINSAT}$. Per ogni formula A , $T \cup \{A\} \in \text{FINSAT}$ o $T \cup \{\neg A\} \in \text{FINSAT}$

Supponiamo per assurdo che $T \cup \{A\} \notin \text{FINSAT}$ e $T \cup \{\neg A\} \notin \text{FINSAT}$.

Vuol dire che esistono $B \stackrel{\text{fin}}{\subseteq} T \cup \{A\}$ e $C \stackrel{\text{fin}}{\subseteq} T \cup \{\neg A\}$ insoddisfacibili.

Dato che per ipotesi $T \in \text{FINSAT}$, sappiamo che $A \in B$, $\neg A \in C$ (altrimenti non potrebbero essere UNSAT, in quanto sarebbero solo $\subseteq T \in \text{FINSAT}$). Possiamo quindi introdurre $\hat{B} = B \setminus \{A\}$ e $\hat{C} = C \setminus \{\neg A\}$.

Sappiamo che l'insieme $\hat{B} \cup \hat{C} \in \text{FINSAT}$, in quanto sottoinsieme finito di T .

Sia α un assegnamento che lo soddisfa. Se $\alpha(A) = 1$, allora soddisfa anche B . Se $\alpha(A) = 0$, soddisfa anche C . In entrambi i casi abbiamo una contraddizione.

1.8.1. Dimostrazione per i linguaggi numerabili

Sia T in un linguaggio numerabile. $T \in \text{FINSAT} \implies T \in \text{SAT}$.

Supponiamo $\mathcal{L} = \{p_1, p_2, \dots\}$ numerabile.

Definiamo una “catena” di teorie come segue:

- $T_0 = T$
- $T_1 = \begin{cases} T_0 \cup \{p_1\} & T_0 \cup \{p_1\} \in \text{FINSAT} \\ T_0 \cup \{\neg p_1\} & T_0 \cup \{\neg p_1\} \in \text{FINSAT} \end{cases}$
- \vdots
- $T_{n+1} = \begin{cases} T_n \cup \{p_{n+1}\} & T_0 \cup \{p_{n+1}\} \in \text{FINSAT} \\ T_n \cup \{\neg p_{n+1}\} & T_0 \cup \{\neg p_{n+1}\} \in \text{FINSAT} \end{cases}$

(aggiungiamo quindi proposizioni una alla volta in modo che T_i resti FINSAT)

(la definizione è ben posta per l'estendibilità di FINSAT)

Avremo quindi $T = T_0 \subseteq T_1 \subseteq T_2 \subseteq \dots$

Definiamo

$$T^* = \bigcup_{n \in \mathbb{N}} T_n$$

Sappiamo che $T^* \in \text{FINSAT}$ perché $\forall X = \{A_1, A_2, \dots, A_k\} \stackrel{\text{fin}}{\subseteq} T^*$, esiste n^* t.c. $X \subseteq T_{n^*}$.

(T è costruito come una catena crescente, quindi ogni suo sottoinsieme finito è un sottoinsieme di uno degli insiemi della catena - quello con "pedice massimo"; per esempio, se $X = \{A_1, A_2\}$ con $A_1 = \{p_1\}$, $A_2 = \{p_3, p_5\}$, avremo $X \subseteq T_5$)

Visto che, per costruzione, $\forall p_n$ vale $(p_n \in T^* \oplus \neg p_n \in T^*)$, possiamo definire un assegnamento:

$$\alpha^*(p_n) = \begin{cases} 1 & p_n \in T^* \\ 0 & p_n \notin T^* (\neg p_n \in T^*) \end{cases}$$

Claim: $\alpha^*(T) = 1$

(avremmo $T \in \text{SAT}$, quindi avremmo finito)

Dobbiamo quindi dimostrare che $\forall A \in T, \alpha^*(A) = 1$.

Abbiamo $A = \{p_{i1}, \dots, p_{ik}\} \in T$.

Introduciamo la notazione: $p_n^* = \begin{cases} p_n & p_n \in T^* (\alpha^*(p_n) = 1) \\ \neg p_n & \neg p_n \in T^* (\alpha^*(p_n) = 0) \end{cases}$

Poiché $A \in T \subseteq T^*$ e $\{p_{i1}^*, \dots, p_{ik}^*\} \subseteq T^*$, abbiamo $A^* = A \cup \{p_{i1}^*, \dots, p_{ik}^*\} = \{A, p_{i1}^*, \dots, p_{ik}^*\} \stackrel{\text{fin}}{\subseteq} T^*$.

Dato che $T^* \in \text{FINSAT}$, $\exists \beta$ t.c. $\beta(A^*) = 1$ (il che può succedere solo se $\beta(A) = 1 \wedge \beta(p_{ij}^*) = 1 \forall j \in [k]$).

Ma, poiché $\beta(p_{ij}^*) = 1 \forall j \in [k]$, notiamo che necessariamente $\beta(p_j) = 1$ se $p_j \in T^*$ e $\beta(p_j) = 0$ se $p_j \notin T^*$. Dunque, notiamo che β e α^* si comportano allo stesso modo per ogni variabile p_{i1}, \dots, p_{ik} .

Da questo (e dall'osservazione a fine pagina 5), poiché p_{i1}, \dots, p_{ik} sono le variabili che compongono A , segue che $\beta(A) = \alpha^*(A)$.

Ma $\beta(A) = 1$ per scelta di β , quindi $\alpha^*(A) = 1$. Visto che possiamo applicare lo stesso ragionamento ad ogni $A \in T$, si ha che $\alpha^*(T) = 1$, ovvero $T \in \text{SAT}$ \square

(ogni proposizione che va verificata, in quanto finita, riguarda solo un sottoinsieme di T , e crea quindi un "bottleneck")

1.8.2. Dimostrazione per i linguaggi arbitrari

Lemma 3: Lemma di Zorn

Sia X un insieme, e $\leq \subseteq X^2$ una relazione di **ordine parziale** (riflessiva, antisimmetrica e transitiva) su X . Definiamo, in X , i concetti di:

- catena $C =$ sottoinsieme di X i cui elementi sono a due a due confrontabili via \leq
- maggiorante = elemento $x \in X$ t.c. $\forall y \in C, y \leq x$

Il **lemma di Zorn** afferma che, se per ogni catena C in X esiste un **maggiorante** in X , allora esiste un

elemento $m \in X$ **massimale**.

Il Lemma di Zorn è una forma dell'Assioma della Scelta (che, informalmente, afferma che quando viene data una collezione di insiemi non vuoti si può sempre costruire un nuovo insieme “scegliendo” un singolo elemento da ciascuno di quelli di partenza).

A noi basta considerare come relazione d'ordine l'inclusione insiemistica \subseteq per la quale l'**unione è un maggiorante**.

Usiamo il Lemma di Zorn per dimostrare (il verso non banale de) il Teorema di Compattezza.

Lemma 4: Lemma di Zorn per famiglie di insiemi

Sia A un insieme e $\mathcal{P}(A)$ il suo insieme delle parti.

Sia $\mathcal{F} \subseteq \mathcal{P}(A)$ una famiglia di sottinsiemi di A .

Se per ogni **catena** \mathcal{C} in \mathcal{F} (i.e., per ogni famiglia di sottinsiemi di A appartenenti a \mathcal{F} i cui elementi sono due a due confrontabili via \subseteq) esiste un **maggiorante** in \mathcal{F} (ossia un sottinsieme S di A in \mathcal{F} tale che per ogni $S' \in \mathcal{C}$ vale $S' \subseteq S$), allora esiste un sottinsieme **massimale** M di A in \mathcal{F} (ossia $M \in \mathcal{F}$ tale che per ogni $S \in \mathcal{F}$, se $M \subseteq S$ allora $S = M$).

Si osserva facilmente che se \mathcal{F} contiene l'**unione** di ogni sua catena allora soddisfa le condizioni di applicabilità del lemma, in quanto l'unione risulta un maggiorante della catena.

Data $T \in \text{FINSAT}$, definiamo $\mathcal{T} = \{\hat{T} \mid T \subseteq \hat{T} \wedge \hat{T} \in \text{FINSAT}\}$, la famiglia di teorie FINSAT che estendono T . Sappiamo che $\mathcal{T} \neq \emptyset$, in quanto contiene almeno T .

Vogliamo verificare che \mathcal{T} verifichi le condizioni per applicare il lemma di Zorn.

Sia $C = \{T_i\}_{i \in \mathbb{N}} \subseteq \mathcal{T}$ una catena crescente. È evidente che $\bigcup_i T_i$ è un maggiorante, e anche che estende T . Sappiamo anche che è FINSAT. Infatti, se consideriamo un qualsiasi sottoinsieme finito di $\bigcup_i T_i$, ogni sua proposizione sarà un elemento di qualche elemento della catena; questo significa che l'insieme stesso è un sottoinsieme di un elemento della catena, ed è quindi FINSAT.

Applicando quindi il lemma di Zorn, otteniamo che \mathcal{T} contiene un massimale T^* , ovvero una teoria tale che:

- $T \subseteq T^*$
- $T^* \in \text{FINSAT}$
- T^* non può essere propriamente esteso mantenendo la condizione di finita soddisfacibilità - ovvero $\forall T' \in \mathcal{T}, T^* \subseteq T' \implies T' = T^*$

In quanto massimale, T^* gode di alcune proprietà:

- (1) data A , non può essere che $\neg A \in T^*$ e $A \in T^*$
- (2) se $A \notin T^*$, necessariamente $\neg A \in T^*$ (altrimenti T^* potrebbe essere estesa con A o $\neg A$ senza perdere la finita soddisfacibilità)
- (3) se $A \in T^*$ e $A \models B$, si ha $B \in T^*$ (T^* è chiuso per conseguenza logica)

(se $B \notin T^*$, si avrebbe $\neg B \in T^*$, ma dato che $A \models B$, si avrebbe $\{A, B\} \subseteq T^* \in \text{UNSAT}$, quindi $T^* \notin \text{FINSAT}$)

Come per la dimostrazione precedente, definiamo un assegnamento

$$\alpha^*(p_n) = \begin{cases} 1 & p_n \in T^* \\ 0 & \neg p_n \in T^* \end{cases}$$

Claim: $\alpha^*(T) = 1$

Dimostrare che α^* soddisfa T^* basta a dimostrare che soddisfa anche T .

Possiamo dimostrare una proprietà più forte: che $\forall A, \alpha(A) = 1 \iff A \in T^*$

Lavoriamo per induzione sulla struttura di A :

■ **caso base:** $A = p_n$ - si ha $\alpha^*(p_n) = 1 \iff p_n \in T^*$

■ **casi induttivi:**

(1) **NOT:** $A = \neg B$

$$\alpha^*(\neg B) = 1 \iff \alpha^*(B) = 0$$

Per l'ipotesi induttiva su B , sappiamo che $\alpha^*(B) = 0 \iff B \notin T^*$. Sfruttando la proprietà di massimalità di T^* (ogni formula o la sua negazione deve appartenere all'insieme):

$$B \notin T^* \iff \neg B \in T^*$$

Pertanto, $\alpha^*(\neg B) = 1 \iff \neg B \in T^*$.

(2) **AND:** $A = B \wedge C$

Dobbiamo mostrare che $\alpha^*(B \wedge C) = 1 \iff (B \wedge C) \in T^*$.

(\Rightarrow) Sia $\alpha^*(B \wedge C) = 1$. Per definizione di valutazione, questo implica $\alpha^*(B) = 1$ e $\alpha^*(C) = 1$. Per l'ipotesi induttiva, abbiamo $B \in T^*$ e $C \in T^*$. Supponiamo per assurdo che $(B \wedge C) \notin T^*$. Per la massimalità, allora $\neg(B \wedge C) \in T^*$. Consideriamo l'insieme finito $\{B, C, \neg(B \wedge C)\} \subseteq T^*$. Questo insieme è insoddisfacibile (UNSAT), il che contraddice il fatto che T^* sia FINSAT (finitamente soddisfacibile). Quindi deve essere $(B \wedge C) \in T^*$.

(\Leftarrow) Sia $(B \wedge C) \in T^*$. Dobbiamo mostrare che $B \in T^*$ e $C \in T^*$. Se $B \notin T^*$, allora $\neg B \in T^*$. L'insieme $\{B \wedge C, \neg B\} \subseteq T^*$ sarebbe UNSAT, impossibile. Quindi $B \in T^*$. Analogamente $C \in T^*$. Per ipotesi induttiva, $\alpha^*(B) = 1$ e $\alpha^*(C) = 1$, quindi $\alpha^*(B \wedge C) = 1$.

(3) **OR:** $A = B \vee C$

Dobbiamo mostrare che $\alpha^*(B \vee C) = 1 \iff (B \vee C) \in T^*$.

(\Rightarrow) Sia $\alpha^*(B \vee C) = 1$. Allora $\alpha^*(B) = 1$ oppure $\alpha^*(C) = 1$. Per ipotesi induttiva, $B \in T^*$ oppure $C \in T^*$. Supponiamo WLOG che $B \in T^*$. L'insieme $\{B, \neg(B \vee C)\}$ è insoddisfacibile. Poiché T^* è FINSAT e $B \in T^*$, non può contenere $\neg(B \vee C)$. Per massimalità, deve contenere $(B \vee C)$.

(\Leftarrow) Sia $(B \vee C) \in T^*$. Supponiamo per assurdo che $\alpha^*(B \vee C) = 0$. Questo implicherebbe $\alpha^*(B) = 0$ e $\alpha^*(C) = 0$. Per ipotesi induttiva, $B \notin T^*$ e $C \notin T^*$. Per massimalità, $\neg B \in T^*$ e $\neg C \in T^*$. Consideriamo l'insieme finito $\{(B \vee C), \neg B, \neg C\} \subseteq T^*$. Questo insieme è chiaramente insoddisfacibile, contraddicendo la proprietà FINSAT di T^* . Quindi l'ipotesi che la valutazione sia 0 è falsa, pertanto $\alpha^*(B \vee C) = 1$.

(4) **IMPLICAZIONE:** $A = B \rightarrow C$

Dobbiamo mostrare che $\alpha^*(B \rightarrow C) = 1 \iff (B \rightarrow C) \in T^*$.

(\Leftarrow) Sia $(B \rightarrow C) \in T^*$. Se $\alpha^*(B \rightarrow C) = 0$, allora $\alpha^*(B) = 1$ e $\alpha^*(C) = 0$. Per ipotesi induttiva, $B \in T^*$ e $C \notin T^*$ (quindi $\neg C \in T^*$). Consideriamo l'insieme finito $\{(B \rightarrow C), B, \neg C\} \subseteq T^*$. Questo insieme è UNSAT([Modus Ponens contraddetto](#)). Poiché T^* è FINSAT, non possiamo avere questa configurazione. Quindi $\alpha^*(B \rightarrow C)$ deve essere 1.

(\Rightarrow) Sia $\alpha^*(B \rightarrow C) = 1$. Questo accade se $\alpha^*(B) = 0$ oppure $\alpha^*(C) = 1$.

- Se $\alpha^*(B) = 0 \implies B \notin T^* \implies \neg B \in T^*$. L'insieme $\{\neg B, \neg(B \rightarrow C)\}$ è UNSAT. Quindi $(B \rightarrow C) \in T^*$.
- Se $\alpha^*(C) = 1 \implies C \in T^*$. L'insieme $\{C, \neg(B \rightarrow C)\}$ è UNSAT. Quindi $(B \rightarrow C) \in T^*$.

Conclusione

Abbiamo mostrato che per ogni A , $\alpha^*(A) = 1 \iff A \in T^*$. Poiché $T \subseteq T^*$ e per ogni $A \in T^*$ vale $\alpha^*(A) = 1$, ne consegue che α^* soddisfa tutte le formule in T . Dunque T è soddisfacibile (SAT).

1.9. Applicazioni del teorema di compattezza

Assegnamenti soddisfacibili

Sia $T = \{A_1, A_2, \dots\}$ una teoria finita numerabile tale che ogni assegnamento soddisfa qualche proposizione in T . Dimostriamo per compattezza che allora $\exists k \in \mathbb{N}$ t.c. $A_1 \vee A_2 \vee \dots \vee A_k \in \text{TAUT}$

Per assurdo:

- supponiamo che ogni assegnamento soddisfi qualche proposizione, ma che nessuna $A_1 \vee \dots \vee A_k$ sia una tautologia
- segue che $\forall j \in \mathbb{N} \exists \alpha$ t.c. $\alpha(A_0 \vee \dots \vee A_j) = 0$
- perciò, $\forall j$ la teoria $\{\neg A_0, \neg A_1, \dots, \neg A_j\}$ è soddisfacibile, ovvero $\hat{T} = \{\neg A_0, \neg A_1, \dots\} \in \text{FINSAT}$
- per compattezza, $\hat{T} \in \text{SAT}$, ovvero $\exists \alpha$ t.c. $\alpha(\hat{T}) = 1$, il che è in contraddizione con l'ipotesi per cui ogni assegnamento soddisfa qualche proposizione. \square

Colorabilità di grafi (Erdős-De Bruijn)

Un grafo $G = (V, E)$ è k -colorabile se esiste una partizione dei vertici in k classi (colori) tale che non esistono archi tra vertici della stessa classe. In altre parole, è possibile assegnare a ogni vertice un colore in modo che vertici adiacenti abbiano colori diversi.

Thm. 6: Teorema di Erdős-De Bruijn

Sia G un grafo infinito. G è k -colorabile se e solo se ogni sottografo di G indotto da un insieme finito di vertici è k -colorabile.

Dimostrazione (per compattezza): Vogliamo dimostrare che se la condizione vale per i sottografi finiti, allora vale per il grafo infinito (l'altra implicazione è banale).

Costruiamo una teoria proposizionale T che descrive la k -colorazione.

Associamo a ogni vertice $v \in V$ e ogni colore $i \in \{1, \dots, k\}$ una variabile $P_{v,i}$. La teoria T è composta da:

- *Esistenza del colore*: ogni vertice ha almeno un colore.

$$P_{v,1} \vee P_{v,2} \vee \cdots \vee P_{v,k} \quad \forall v \in V$$

- *Unicità del colore*: un vertice non ha più colori contemporaneamente.

$$\neg(P_{v,i} \wedge P_{v,j}) \quad \forall v \in V, i \neq j$$

- *Vincoli di adiacenza*: vertici collegati hanno colori diversi.

$$\neg(P_{v,i} \wedge P_{w,i}) \quad \forall \{v, w\} \in E, \forall i$$

Se un assegnamento α soddisfa T , definisce una colorazione valida ponendo $c(v) = i \iff \alpha(P_{v,i}) = 1$.

Dobbiamo mostrare che **ogni sottoinsieme finito $T_0 \subseteq T$ è soddisfacibile**.

- sia T_0 un sottoinsieme finito di T .
- sia V_0 l'insieme dei vertici menzionati nelle formule di T_0 . Poiché T_0 è finito, V_0 è finito.
- consideriamo il sottografo indotto da V_0 . Per ipotesi del teorema, esso è k -colorabile tramite una colorazione c .
- definiamo l'assegnamento α basato su c :

$$\alpha(P_{v,i}) = 1 \iff v \in V_0 \wedge c(v) = i$$

- poiché c è una colorazione valida sul sottografo finito, α soddisfa tutte le formule in T_0 .

Poiché $T \in \text{FINSAT}$, per il Teorema di Compattezza $T \in \text{SAT}$. Esiste quindi un assegnamento che soddisfa T , il quale induce una k -colorazione valida per tutto G . \square

Estensione di ordini parziali

Ogni ordine parziale può essere esteso a un ordine totale.

Thm. 7

Per ogni ordine parziale $(X, <)$ esiste un ordine totale (X, \prec) che estende $<$, ovvero tale che:

$$\text{se } x < y \text{ allora } x \prec y \quad \forall x, y \in X$$

Dimostrazione (per compattezza): Assumiamo come vero il teorema per insiemi finiti e usiamo questo fatto per dimostrare il caso infinito.

Il nostro obiettivo è trovare un assegnamento di verità che descriva la relazione d'ordine totale \prec . Introduciamo le variabili proposizionali $p_{x,y}$ con il significato di “ $x \prec y$ ”.

La teoria è composta dalle seguenti formule:

- Per ogni $x, y, z \in X$:
 - *Irriflessività*: $\neg p_{x,x}$
 - *Asimmetria*: $p_{x,y} \rightarrow \neg p_{y,x}$
 - *Transitività*: $(p_{x,y} \wedge p_{y,z}) \rightarrow p_{x,z}$
 - *Totalità*: $p_{x,y} \vee p_{y,x}$ (per $x \neq y$)

- Dobbiamo garantire che il nuovo ordine rispetti quello vecchio. Per ogni coppia $x, y \in X$ tale che $x < y$ nell'ordine originale, aggiungiamo la formula:

$$p_{x,y}$$

Se T è soddisfacibile, un modello α definisce l'ordine totale: $x \prec y \iff \alpha(p_{x,y}) = 1$.

Dimostriamo che ogni sottoinsieme finito $T_0 \subseteq T$ è soddisfacibile.

- Sia $T_0 \subseteq T$ un insieme finito di formule.
- Sia X_0 l'insieme dei vertici $x, y \in X$ che compaiono come indici nelle variabili $p_{x,y}$ presenti in T_0 . Poiché T_0 è finito, anche X_0 è finito.
- Consideriamo la restrizione dell'ordine parziale originale $<$ all'insieme X_0 . Otteniamo un ordine parziale finito $(X_0, <|_{X_0})$.
- Poiché X_0 è finito, sappiamo che esiste un ordine totale \prec_0 su X_0 che estende $<|_{X_0}$.
- Usiamo \prec_0 per definire un assegnamento α per le variabili in T_0 :

$$\alpha(p_{x,y}) = 1 \iff x \prec_0 y$$

- Questo assegnamento soddisfa T_0 perché è un ordine totale, e \prec_0 è un'estensione dell'ordine originale (se $x < y$ in X_0 , allora $x \prec_0 y$, quindi $\alpha(p_{x,y}) = 1$).

Poiché ogni sottoinsieme finito T_0 è soddisfacibile, per il Teorema di Compattezza l'intera teoria T è soddisfacibile. Esiste dunque un modello globale che definisce un ordine totale \prec su tutto X che estende l'ordine parziale di partenza $<$. \square

1.9.1. Lemma di König

Thm. 8: Lemma di König

Ogni albero binario infinito A contiene un ramo infinito.

Dimostrazione (per compattezza): Vogliamo dimostrare l'esistenza di un ramo infinito costruendo una teoria proposizionale che lo descriva.

Fissiamo il linguaggio contenente una variabile p_σ per ogni stringa $\sigma \in A$. L'intuizione è che p_σ è vera se il nodo σ fa parte del ramo infinito. La teoria T_A è composta da tre tipi di formule:

- **Esistenza (il ramo non muore):** per ogni livello $n \in \mathbb{N}$, il ramo deve contenere almeno un nodo di quel livello. Se $\sigma_1, \dots, \sigma_k$ sono tutti i nodi di A di lunghezza n :

$$p_{\sigma_1} \vee p_{\sigma_2} \vee \dots \vee p_{\sigma_k}$$

(poiché l'albero è binario, ogni livello ha un numero finito di nodi, quindi la disunione è finita).

- **Unicità (il ramo non si divide):** per ogni livello n , il ramo contiene al massimo un nodo.

$$\neg(p_\sigma \wedge p_\tau) \quad \text{per ogni coppia } \sigma, \tau \in A \text{ di livello } n \text{ con } \sigma \neq \tau$$

- **Connessione (chiusura verso l'alto):** se un nodo fa parte del ramo, anche il suo predecessore deve farne parte.

$$p_\sigma \rightarrow p_\tau \quad \text{se } \tau \text{ è un segmento iniziale (prefisso) di } \sigma$$

Dobbiamo mostrare che ogni sottoinsieme finito $T_0 \subseteq T_A$ è soddisfacibile.

- Sia T_0 un sottoinsieme finito di T_A .
- Consideriamo tutte le variabili p_σ che compaiono in T_0 . Poiché sono in numero finito, esiste una lunghezza massima N tra le stringhe menzionate.
- Poiché l'albero A è infinito per ipotesi, deve necessariamente contenere nodi di livello N (altrimenti avrebbe un numero finito totale di nodi).
- Scegliamo un nodo $\bar{\sigma} \in A$ di livello N .
- Consideriamo il percorso finito dalla radice fino a $\bar{\sigma}$. Definiamo l'assegnamento α tale che:

$$\alpha(p_s) = 1 \iff s \text{ è un prefisso di } \bar{\sigma}$$

- Questo assegnamento soddisfa T_0 . Infatti:
 - Per ogni livello $n \leq N$, c'è esattamente un nodo vero (il prefisso di lunghezza n di $\bar{\sigma}$), soddisfacendo esistenza e unicità.
 - Le implicazioni di connessione sono soddisfatte perché stiamo prendendo un ramo dell'albero.

Poiché $T_A \in \text{FINSAT}$, allora $T_A \in \text{SAT}$ per Compattezza. Sia α un modello di T_A . L'insieme $R = \{\sigma \in A : \alpha(p_\sigma) = 1\}$ definisce un ramo infinito in A , in quanto interseca ogni livello in un unico punto e rispetta l'ordinamento. \square

1.10. Decidibilità

Dato il potere espressivo della logica proposizionale, è naturale chiedersi se sia possibile automatizzare la risposta alla domanda “ $T \models A$ ”.

Se $T = \{A_1, \dots, A_n\}$ è una **teoria finita**, la risposta è banalmente “sì”, in quanto sappiamo che $T \models A \iff (A_1 \wedge \dots \wedge A_n) \rightarrow A \in \text{TAUT}$ (il che è facilmente verificabile tramite tavole di verità).

Def. 9: Decidibilità

Dato uno spazio X di possibili input, chiamiamo un *problema* un qualsiasi sottoinsieme $S \subseteq X$.

Diciamo che S è **algoritmamente decidibile** se esiste un algoritmo tale che $\forall x \in X$, se $x \in S$, l'algoritmo su input x termina in tempo finito e risponde “sì”, e se $x \notin S$, l'algoritmo su input x termina in tempo finito e risponde “no”.

Se invece T è una teoria **infinita numerabile**, potremmo usare il *teorema di compattezza* per fare un ragionamento del genere:

- Sappiamo che $T \models A \iff \exists T_0 \stackrel{\text{fin}}{\subseteq} T \text{ t.c. } T_0 \models A$
- Indicando con $Fin(T)$ l'insieme dei sottoinsiemi finiti di T , sappiamo che $Fin(T)$ è numerabile (in quanto T lo è).
- Se potessimo quindi produrre algoritmicamente un'enumerazione di $Fin(T)$ del tipo S_1, S_2, S_3, \dots , poiché, grazie al teorema di compattezza, sappiamo che $\exists i \in \mathbb{N} \text{ t.c. } S_i \models A$, potremmo seguire questa procedura:

partendo da $i = 1$, ci chiediamo se $S_i \models A$. Poiché S_i è finito, si può rispondere algoritmicamente. Se la risposta è “sì”, terminiamo la procedura e rispondiamo “sì”. Altrimenti, ripetiamo con $i + 1$.

Se l'enumerazione di $Fin(T)$ si può produrre algoritmicamente, allora tutta la procedura è algoritmica. Notiamo però che, mentre nel caso in cui $T \models A$ sicuramente l'algoritmo terminerà e darà la risposta esatta, nel caso in cui $T \not\models A$, esso non terminerà mai (visto che T è infinita).

Chiamiamo questo tipo di problema semi-decidibile.

Def. 10: Problema semi-decidibile

Dato uno spazio ambiente X e un problema $S \subseteq X$, diciamo che S è **semi-decidibile** se esiste un algoritmo tale che $\forall x \in X$, se $x \in S$, l'algoritmo (su input x) termina e risponde “sì”; se invece $x \notin S$, l'algoritmo (su input x) continua all'infinito (*diverge*).

(equivalente al concetto di “problema Turing-riconoscibile”)

Def. 11: Problema computabilmente enumerabile

Un insieme infinito per cui esiste una procedura algoritmica di enumerazione di tutti e soli i suoi elementi è detto **computabilmente enumerabile**.

(Notiamo che $\neg(\text{numerabile} \rightarrow \text{computabilmente enumerabile})$)

Thm. 9

Se T è computabilmente enumerabile, allora il problema $T \models A$ è semi-decidibile.

Notiamo quindi che, se lo spazio X dei possibili input è computabilmente enumerabile, allora:

- ogni problema decidibile è anche semi-decidibile (questo sempre)
- un problema è semi-decidibile se e solo se è computabilmente numerabile

Possiamo stabilire delle proprietà di T che ci garantiscano la decidibilità? La risposta è sì.

Consideriamo la procedura introdotta poco fa ed estendiamola in questo modo:

- ad ogni passo, controlliamo non solo $S_i \models A$, ma anche $S_i \models \neg A$
- se $S_i \models A$, terminiamo e rispondiamo “sì”; se $S_i \models \neg A$, terminiamo e rispondiamo “no”

Escludiamo le teorie per cui si ha $T \models A \wedge T \models \neg A$, in quanto sono “**incoerenti**” (ed insoddisfacibili).

Ci restano quindi tre casi:

- (1) **Caso 1:** $T \models A$ e $T \not\models \neg A$: la procedura applicata a A termina e risponde affermativamente mentre la procedura applicata a $\neg A$ diverge. Possiamo concludere che $T \models A$.
- (2) **Caso 2:** $T \not\models A$ e $T \models \neg A$: la procedura applicata a A diverge e la procedura applicata a $\neg A$ termina e risponde affermativamente. Possiamo comunque concludere che $T \models \neg A$. Se T non è insoddisfacibile, non può essere che $T \models A$. Dunque possiamo concludere e rispondere che $T \models A$.
- (3) **Caso 3:** $T \not\models A$ e $T \not\models \neg A$: La procedura diverge quando viene applicata sia ad A che a $\neg A$. Questo caso esiste, ma vogliamo escluderlo.

Def. 12: Teoria semanticamente completa

Una teoria T è detta **semanticamente completa** se $\forall A$ nel linguaggio di T , vale esattamente una tra $T \models A$ e $T \models \neg A$.

Da questo possiamo derivare che:

Thm. 10

Se T è computabilmente enumerabile e semanticamente completa, allora $T \models A?$ è decidibile algoritmamente $\forall A$.

Notiamo che le proprietà seguenti sono equivalenti:

- (1) T è semanticamente completa.
- (2) Per ogni formula A , vale $T \models A \iff T \not\models \neg A$.
- (3) T è soddisfacibile e per ogni formula A se $T \not\models A$ allora $T \models \neg A$.
- (4) T ha un unico modello.
- (5) Per ogni formula A, B vale $T \models A \vee B$ se e solo se $T \models A$ oppure $T \models B$.
- (6) Per ogni formula A, B vale $T \not\models A \rightarrow B$ se e solo se $T \models A$ e $T \models \neg B$.

Calcolo Proposizionale

Nota: sintassi (\vdash) vs semantica (\vDash)

È fondamentale distinguere tra due piani che idealmente coincidono, ma sono concettualmente diversi:

- *Semantica* ($T \vDash A$): riguarda il significato dei simboli e la loro verità - ci si chiede se una formula sia vera in tutti i “mondi possibili” (es. analizzando le tavole di verità)
- *Sintassi* ($T \vdash A$): riguarda la forma e la deduzione - ci si chiede se una formula sia derivabile dagli assiomi rispettando le regole del sistema di prova interessato (non conta il significato dei simboli, ma solo la loro manipolazione meccanica tramite le regole di inferenza)

L’obiettivo del *Calcolo Proposizionale* è costruire una macchina sintattica formata da assiomi e regole che sia capace di catturare esattamente tutte le verità semantiche (teoremi di correttezza e completezza).

Una dimostrazione rigorosa è una successione ordinata e finita di asserzioni, ognuna delle quali può essere giustificata richiamandosi a una verità assunta come ipotesi (assioma), o a una regola di ragionamento corretta che permette di ottenerla da altre proposizioni.

La regola che utilizziamo nel nostro sistema di dimostrazioni (“alla Hilbert”) è il **Modus Ponens**: da $X \wedge (X \rightarrow Y)$ segue Y .

Lo scriviamo in questo modo:
$$\frac{X \quad X \rightarrow Y}{Y}$$

Def. 13: Dimostrazione

Una **dimostrazione** / deduzione è una *successione finita* F_1, \dots, F_k di proposizioni t.c., $\forall i \in [k]$:

- F_i è un’istanza di un assioma, oppure
- F_i si ottiene da due formule precedenti tramite regole di inferenza

Nel nostro sistema (in cui limitiamo il linguaggio ai connettivi \neg e \rightarrow), scegliamo come assiomi:

- (1) $X \rightarrow (Y \rightarrow X)$
 - (2) $(X \rightarrow (Y \rightarrow Z)) \rightarrow ((X \rightarrow Y) \rightarrow (X \rightarrow Z))$
- (se X implica $Y \rightarrow Z$, allora $X \rightarrow Y$ implica $X \rightarrow Z$ (una sorta di transitività))

$$(3) (\neg Y \rightarrow \neg X) \rightarrow (X \rightarrow Y)$$

Abbiamo scelto questo sistema perché vogliamo la completezza rispetto alla conseguenza logica. Vogliamo quindi che $\models A \iff A$ è dimostrabile da queste regole di inferenza e ipotesi in T .

Def. 14: Dimostrazione nel calcolo proposizionale

Una **dimostrazione** / deduzione di A da T nel C.P. è una *successione finita* F_1, \dots, F_k di proposizioni t.c.:

- $F_k = A$
- $\forall i \in [k]$, si può avere che:
 - F_i è un'istanza di **assioma**
 - $F_i \in T$
 - $\exists p, q < i \text{ t.c. } \frac{F_q \quad F_p = F_q \rightarrow F_i}{F_i}$
(è un'istanza del M.P.)

Def. 15: Teorema

A è un teorema se:

$$\vdash A$$

ovvero A è dimostrabile a partire “semplicemente” dagli assiomi.

Thm. 11: Teorema di Correttezza

Tutti i teoremi sono tautologie.

$$\vdash A \implies \models A$$

Possiamo estendere il teorema facilmente anche a:

$$T \vdash A \implies T \models A$$

La dimostrazione è semplice: $\vdash A$ significa che A è dimostrabile a partire dagli assiomi logici (che sono verità logiche), e il Modus Ponens preserva le verità logiche (e il discorso è facilmente estendibile per $T \vdash A \implies T \models A$).

Se scriviamo $Teor(T) = \{A : T \vdash A\}$, la Correttezza si esprime insiemisticamente in questo modo:

$$Teor(T) \subseteq Cons(T)$$

ovvero, il nostro sistema permette di derivare formalmente dalle ipotesi di T solo *conseguenze logiche* di T .

2.1. Teorema di completezza

Thm. 12: Teorema di completezza

Il teorema di completezza sancisce l'equivalenza tra la nozione semantica di conseguenza logica e quella sintattica di derivabilità formale:

$$\begin{aligned}\vdash A &\iff \models A \\ T \vdash A &\iff T \models A\end{aligned}$$

idea di dimostrazione ($T \models A \implies T \vdash A$):

Dal teorema di compattezza sappiamo che

$$\begin{aligned}T \models A &\iff \exists \{A_1, \dots, A_n\} \subseteq T \text{ t.c. } A_1, \dots, A_n \models A \\ &\iff \models (A_1 \rightarrow (A_2 \rightarrow \dots (A_n \rightarrow A) \dots))\end{aligned}$$

Dimostreremo che tutte le tautologie sono teoremi del calcolo proposizionale, e che quindi

$$\exists \{A_1, \dots, A_n\} \subseteq T \text{ t.c. } \vdash (A_1 \rightarrow (A_2 \rightarrow \dots (A_n \rightarrow A) \dots))$$

Da questo vogliamo ottenere $T \vdash A$.

Lo faremo verificando che $\vdash (A_1 \rightarrow (A_2 \rightarrow \dots (A_n \rightarrow A) \dots)) \iff A_1, \dots, A_n \vdash A$.

2.1.1. Teorema di Deduzione

Il passaggio chiave per collegare la soddisficiabilità alla dimostrabilità è il Teorema di Deduzione, che ci permette di spostare le premesse all'interno della formula sotto forma di implicazione.

Thm. 13: Teorema di Deduzione [Herbrand, 1930]

Sia T un insieme di formule e A, B due formule. Allora:

$$T, B \vdash A \iff T \vdash (B \rightarrow A)$$

dim

Dobbiamo dimostrare la doppia implicazione.

Direzione 1: $T \vdash (B \rightarrow A) \implies T, B \vdash A$

Questa direzione è immediata.

- (1) Sia (G_1, \dots, G_m) una derivazione di $B \rightarrow A$ a partire da T .
- (2) Aggiungiamo B come nuova ipotesi (poiché ora lavoriamo in $T \cup \{B\}$).
- (3) Avendo B e $B \rightarrow A$, per Modus Ponens otteniamo A .
- (4) Dunque $T, B \vdash A$.

Direzione 2: $T, B \vdash A \implies T \vdash (B \rightarrow A)$

Sia (F_1, \dots, F_n) una derivazione di A (quindi $F_n = A$) a partire dalle premesse $T \cup \{B\}$. Dobbiamo dimostrare per induzione sull'indice i (lunghezza della derivazione) che $T \vdash (B \rightarrow F_i)$ per ogni $i = 1, \dots, n$.

Caso base ($i = 1$): F_1 è l'inizio della derivazione. Ci sono tre casi:

- **Caso 1:** F_1 è un assioma logico oppure $F_1 \in T$.

In questo caso $T \vdash F_1$ (senza bisogno di B). Utilizziamo l'assioma logico $F_1 \rightarrow (B \rightarrow F_1)$. Per Modus Ponens, otteniamo $T \vdash B \rightarrow F_1$.

- **Caso 2:** F_1 è la formula B .

Dobbiamo dimostrare $T \vdash B \rightarrow B$. Sappiamo che $\vdash B \rightarrow B$ è un teorema logico. Quindi a maggior ragione $T \vdash B \rightarrow B$.

Passo induttivo ($i > 1$): Supponiamo che la tesi valga per tutti i passi $k < i$. Analizziamo F_i :

- Se F_i è un assioma, un elemento di T o B , si procede come nel caso base.

- **Caso 3:** F_i è ottenuto per **Modus Ponens**.

Significa che esistono $m, l < i$ tali che F_l è la formula $F_m \rightarrow F_i$. Per ipotesi induttiva sappiamo che:

- 1) $T \vdash B \rightarrow F_m$
- 2) $T \vdash B \rightarrow (F_m \rightarrow F_i)$ (poiché $F_l = F_m \rightarrow F_i$)

Utilizziamo l'Assioma 2 (distributività dell'implicazione):

$$(B \rightarrow (F_m \rightarrow F_i)) \rightarrow ((B \rightarrow F_m) \rightarrow (B \rightarrow F_i))$$

Applicando il Modus Ponens due volte (usando le ipotesi induttive 1 e 2), concludiamo:

$$T \vdash B \rightarrow F_i$$

Poiché $F_n = A$, l'induzione prova che $T \vdash B \rightarrow A$.

2.1.2. Completezza Semplice

L'obiettivo di questa sezione è dimostrare che le verità logiche (tautologie) sono dimostrabili nel calcolo proposizionale.

$$\models B \implies \vdash B$$

Per farlo, utilizziamo un lemma fondamentale che collega la verità semantica (valutazioni α) alla dimostrabilità sintattica.

Lemma 5

Sia B una formula contenente le variabili proposizionali p_1, \dots, p_k . Sia α un assegnamento di verità. Definiamo la formula B^α come:

$$B^\alpha = \begin{cases} B & \text{se } \alpha(B) = 1 \\ \neg B & \text{se } \alpha(B) = 0 \end{cases}$$

(e analogamente per le variabili p_j^α). Allora vale sempre:

$$p_1^\alpha, \dots, p_k^\alpha \vdash B^\alpha$$

Si procede per **induzione** sul numero n di connettivi logici (\neg, \rightarrow) in B .

Base ($n = 0$): B è una variabile atomica p_i . La tesi diventa $p_i \vdash p_i$ (se $\alpha(p_i) = 1$) oppure $\neg p_i \vdash \neg p_i$ (se $\alpha(p_i) = 0$). Entrambe sono banalmente vere.

Passo Induttivo ($n > 0$): Supponiamo il lemma vero per formule con meno connettivi. Analizziamo la forma di B :

■ **Caso $B = \neg C$:**

- Se $\alpha(C) = 0$, allora $\alpha(B) = 1$. Per ipotesi induttiva $\Gamma \vdash C^\alpha$ cioè $\Gamma \vdash \neg C$. Ma B^α è B (cioè $\neg C$). Quindi $\Gamma \vdash B^\alpha$.
- Se $\alpha(C) = 1$, allora $\alpha(B) = 0$. Per ipotesi induttiva $\Gamma \vdash C$. Dobbiamo dimostrare $\Gamma \vdash \neg B$ (cioè $\neg\neg C$). Usiamo il teorema $\vdash C \rightarrow \neg\neg C$.

■ **Caso $B = C \rightarrow D$:** Si analizzano i valori di verità di C e D . Ad esempio, se $\alpha(C) = 1$ e $\alpha(D) = 0$, allora $\alpha(B) = 0$. Per ipotesi induttiva abbiamo $\Gamma \vdash C$ e $\Gamma \vdash \neg D$. Dobbiamo dimostrare $\Gamma \vdash \neg(C \rightarrow D)$. Questo segue dal teorema $C \rightarrow (\neg D \rightarrow \neg(C \rightarrow D))$.

Thm. 14: Teorema di Completezza Semplice

Se B è una tautologia, allora $\vdash B$.

Sia B una tautologia con variabili p_1, \dots, p_k . Poiché è una tautologia, per *ogni* assegnamento α vale $\alpha(B) = 1$, quindi $B^\alpha = B$. Dal Lemma precedente segue che per ogni assegnamento α :

$$p_1^\alpha, \dots, p_k^\alpha \vdash B$$

Consideriamo due assegnamenti α e β che differiscono solo per il valore di p_k :

- $\alpha(p_k) = 1 \implies p_1^\alpha, \dots, p_{k-1}^\alpha, p_k \vdash B$
- $\beta(p_k) = 0 \implies p_1^\alpha, \dots, p_{k-1}^\alpha, \neg p_k \vdash B$

Applicando il Teorema di Deduzione a entrambi:

$$\Gamma \vdash p_k \rightarrow B \quad \text{e} \quad \Gamma \vdash \neg p_k \rightarrow B$$

(dove $\Gamma = \{p_1^\alpha, \dots, p_{k-1}^\alpha\}$).

Utilizzando il teorema noto $(A \rightarrow B) \rightarrow ((\neg A \rightarrow B) \rightarrow B)$, otteniamo:

$$p_1^\alpha, \dots, p_{k-1}^\alpha \vdash B$$

Abbiamo eliminato la dipendenza da p_k . Iterando il procedimento per $k - 1, k - 2, \dots, 1$, eliminiamo tutte le premesse e otteniamo infine:

$$\vdash B$$

2.1.3. Conclusione del Teorema di Completezza

Grazie al Teorema di Deduzione, possiamo ora completare la strategia descritta all'inizio:

- (1) **Dal Teorema di Compattezza:** $T \models A \implies \exists \{A_1, \dots, A_n\} \subseteq T$ tale che $A_1, \dots, A_n \models A$.

(2) Questo equivale a dire che $\models (A_1 \rightarrow (\dots \rightarrow A) \dots)$ è una tautologia.

(3) Per la **Completezza Semplice** (le tautologie sono teoremi), abbiamo:

$$\vdash (A_1 \rightarrow (A_2 \rightarrow \dots (A_n \rightarrow A) \dots))$$

(4) Applicando n volte il **Teorema di Deduzione** (nella direzione \Leftarrow), “scarichiamo” le premesse una alla volta:

$$A_1, A_2, \dots, A_n \vdash A$$

(5) Poiché $\{A_1, \dots, A_n\} \subseteq T$, per la proprietà di **monotonía** concludiamo:

$$T \vdash A$$

3.1. Introduzione

Abbiamo visto come la logica proposizionale ci permette di modellare alcuni tipi di problemi facilmente. Tuttavia, per altri tipi di problemi è necessario un maggiore potere espressivo.

Def. 16: Struttura relazionale

Una struttura relazionale è una tupla $\mathcal{A} = (A, R_1, \dots, R_n, f_1, \dots, f_n, c_1, \dots, c_n)$, dove:

- A rappresenta un insieme di elementi (tipicamente non vuoto) (*dominio*)
- R_1, \dots, R_n sono *relazioni* su A (anche unarie), ossia suoi sottoinsiemi
- f_1, \dots, f_n sono *funzioni* su A
- c_1, \dots, c_k sono *costanti* su A

Esempio: un esempio di struttura è quella di gruppo.

Un gruppo è definito da un insieme A (di elementi del gruppo), un elemento $e \in A$ (l'elemento neutro) e un'operazione binaria $\circ : A \times A \rightarrow A$, che soddisfa le seguenti proprietà:

- (1) Per ogni $a \in A$, $a \circ e = e \circ a = a$
- (2) Per ogni $a \in A$ esiste un $b \in A$ tale che $a \circ b = b \circ a = e$
- (3) Per ogni $a, b, c \in A$: $a \circ (b \circ c) = (a \circ b) \circ c$.

(Un gruppo è inoltre abeliano se soddisfa: per ogni $a, b \in A$ si ha $a \circ b = b \circ a$.)

Def. 17: Linguaggio predicativo

Un **linguaggio predicativo** \mathcal{L} è una collezione (finita o infinita) di simboli di tre tipi:

- simboli di *relazioni*, con la loro arietà
- simboli di *funzioni*, con la loro arietà
- simboli di *costanti* (di arietà 0)

Inoltre, assumiamo sempre un insieme numerabile di *variabili* v_1, v_2, \dots

Esempio: per continuare con l'esempio di sopra, un linguaggio adeguato per la teoria dei gruppi è $\mathcal{L}_G = \{\cdot, e\}$, dove \cdot è un simbolo di funzione binaria per l'operazione di gruppo, e e è una costante per l'elemento neutro.

Nota bene

Nella logica dei predicati, si può quantificare solo sugli *elementi* delle strutture e non sulle strutture stesse (si chiama, per questo, Logica di Primo Ordine o FOL, First-Order Logic).

Per quantificare sulle strutture, esiste la Logica di Secondo Ordine (non trattata in questo corso).

Def. 18: Termini

I **termini** sono ottenuti partendo dalle variabili e dalle costanti e chiudendo sotto applicazione dei simboli di funzione.

Un termine che non contiene variabili è un **termine chiuso**.

Per formulare proposizioni nel linguaggio \mathcal{L} , si usano i seguenti simboli logici:

- **connettivi:** $\wedge, \vee, \rightarrow, \neg$
- **quantificatori:** \exists, \forall
- il simbolo di **eguaglianza:** $=$

Def. 19: Formule

Una **formula atomica** è una stringa del tipo $R(t_1, \dots, t_k)$, dove R è un simbolo di relazione di arità k e t_1, \dots, t_k sono termini, oppure una stringa del tipo $t = s$, dove t e s sono termini.

Le **formule** sono ottenute partendo dalle formule atomiche e chiudendo sotto connettivi proposizionali e quantificatori universali ed esistenziali.

Nelle formule $(\forall v)F$ e $(\exists v)F$, F è detto il **dominio** (o *scope*) del quantificatore v la variabile quantificata. Se v non occorre in F , possiamo identificare le due formule quantificate con F .

Un'occorrenza di una variabile x in una formula F è **vincolata** se e solo se:

- l'occorrenza di x è la variabile quantificata di un quantificatore, oppure
- l'occorrenza di x è nel dominio di un quantificatore con variabile quantificata

Tutte le altre occorrenze di x sono dette **libere**.

Un **enunciato** è una formula senza variabili libere.

Se F è una formula e x_1, \dots, x_n sono variabili distinte, indichiamo con $F(x_1, \dots, x_n)$ il fatto che le variabili libere di F sono contenute nell'insieme $\{x_1, \dots, x_n\}$.

Def. 20: Termine libero

Un termine t è **libero per una variabile** v in una formula F se nessuna occorrenza libera di v in F è nel dominio di un quantificatore $\forall y$ o $\exists y$ con y una variabile in t .

(ovvero, se si sostituisce v con t nella formula, nessuna delle variabili di t viene legata da un quantificatore presente in F)

Per esempio, se F è $\exists y(x = y + y)$, nessun termine contenente y è libero per x in F .

3.2. Semantica

Vogliamo poter definire il concetto di **verità logica** nella logica dei predicati (analogo a quello di tautologia nella logica proposizionale). A tale scopo, ci serve definire la nozione di verità di una formula della logica predicativa.

La verità di una formula della logica predicativa dipende dalla scelta dell'ambiente in cui decidiamo di interpretare i simboli del linguaggio. Un tale ambiente è detto **struttura**.

Def. 21: Struttura per un linguaggio \mathcal{L}

Dato un linguaggio \mathcal{L} , una **struttura** (o interpretazione) \mathcal{A} per il linguaggio \mathcal{L} consiste di:

- un insieme A non vuoto, detto **dominio**
- per ogni simbolo R_i di arietà d , una relazione di arietà d sul dominio A , che denotiamo con $R_i^{\mathcal{A}}$
dove $R_i^{\mathcal{A}} \subseteq A^d$
- per ogni simbolo f_j di arietà d , una funzione a d argomenti sul dominio A , che denotiamo con $f_j^{\mathcal{A}}$
dove $f_j^{\mathcal{A}} : A^d \rightarrow A$
- per ogni $k \in K$, un elemento di A che denotiamo con $c_k^{\mathcal{A}}$

Un **assegnamento** α in \mathcal{A} è una mappa che associa ad ogni variabile un elemento di A , i.e.:

$$\alpha : \{v_n : n \in \mathbb{N}\} \rightarrow A$$

Un assegnamento si estende in modo univoco ai termini ponendo:

- $\alpha(c) = c^{\mathcal{A}}$
- $\alpha(f(t_1, \dots, t_k)) = f^{\mathcal{A}}(\alpha(t_1), \dots, \alpha(t_k))$.

Indichiamo con $\alpha(x_a)$ l'assegnamento che differisce da α solo perché associa l'elemento a alla variabile x .

Def. 22: Soddisfazione

Definiamo la relazione $\mathcal{A} \models F[\alpha]$, che significa “la formula F è soddisfatta nella struttura \mathcal{A} relativamente all'assegnamento α ”

La definiamo per induzione sulla complessità di F (semantica Tarskyana):

- $\mathcal{A} \models R(t_1, \dots, t_k)[\alpha]$ se e solo se $(\alpha(t_1), \dots, \alpha(t_k)) \in R^{\mathcal{A}}$, dove t_1, \dots, t_k sono termini e R è un simbolo di relazione di varietà k .
- $\mathcal{A} \models (t = s)[\alpha]$ se e solo se $\alpha(t) = \alpha(s)$, dove t e s sono termini.
- $\mathcal{A} \models \neg G[\alpha]$ se e solo se non vale $\mathcal{A} \models G[\alpha]$ ($\mathcal{A} \models \neg G[\alpha] \iff \mathcal{A} \not\models G[\alpha]$).
- $\mathcal{A} \models (G \wedge H)[\alpha]$ se e solo se $\mathcal{A} \models G[\alpha]$ e $\mathcal{A} \models H[\alpha]$.
- $\mathcal{A} \models (G \vee H)[\alpha]$ se e solo se $\mathcal{A} \models G[\alpha]$ o $\mathcal{A} \models H[\alpha]$.
- $\mathcal{A} \models (G \rightarrow H)[\alpha]$ se e solo se: se $\mathcal{A} \models G[\alpha]$ allora $\mathcal{A} \models H[\alpha]$.
- $\mathcal{A} \models (\exists xG)[\alpha]$ se e solo se esiste $a \in \mathcal{A}$ tale che $\mathcal{A} \models G[\alpha(x/a)]$.
- $\mathcal{A} \models (\forall xG)[\alpha]$ se e solo se per ogni $a \in \mathcal{A}$ vale $\mathcal{A} \models G[\alpha(x/a)]$.

Notiamo quindi che una struttura decide univocamente lo stato di verità di qualsiasi formula del suo linguaggio.

Ovvero $\text{Th}(\mathcal{A})$, l'insieme di tutte le frasi vere in una specifica struttura \mathcal{A} , è una teoria completa.

Osservazione

Il fatto che valga $\mathcal{A} \models F[\alpha]$ dipende solo dai valori di α sulle variabili libere che appaiono in F . Quindi, come già visto in maniera simile per la logica proposizionale, se α e β sono due assegnamenti che coincidono sui valori assegnati alle variabili x_1, \dots, x_n , allora $\mathcal{A} \models F[\alpha] \iff \mathcal{A} \models F[\beta]$.

Da questo segue che, se F è un **enunciato**, allora $\mathcal{A} \models F[\alpha]$ vale **per tutti gli assegnamenti o per nessuno**.

Def. 23: Soddisfabilità, Validità

Se $\exists \alpha \text{ t.c. } \mathcal{A} \models E[\alpha]$, diciamo che α *soddisfa* un enunciato E in \mathcal{A} . In questo caso, E è detto **soddisfacibile** in \mathcal{A} .

Diciamo che una formula F è **vera** in una struttura se è *soddisfatta da tutti gli assegnamenti* in quella struttura.

Una formula F è vera in una struttura se e solo se l'enunciato

$$\forall x_1, \dots, \forall x_n F(x_1, \dots, x_n)$$

(dove x_1, \dots, x_n sono tutte e sole le variabili libere di F) è vero nella struttura.

Un enunciato E è **valido** se è vero in tutte le strutture (ossia se $\forall \mathcal{A}, \mathcal{A} \models E$). In tal caso, scriviamo $\models E$.

Dualmente, E è *insoddisfacibile* se non esiste una struttura in cui è vero.

Si può estendere il concetto di verità logica anche alle formule predicative. Per esempio, la formula $\forall x R(x) \vee \neg \forall x R(x)$ (ottenuta da $p \vee \neg p$ sostituendo p) è soddisfatta in ogni struttura che possieda un'interpretazione per R .

Questo tipo di formule prendono il nome di *istanze di tautologie proposizionali*.

3.3. Teorie

Dato un linguaggio \mathcal{L} , ci interessa introdurre questi insiemi degni di nota:

- **Teoria** - una teoria è un insieme T di enunciati in \mathcal{L}
- **Modello** - un modello di una teoria è una struttura per \mathcal{L} che *soddisfa tutti gli elementi di T* .
scriviamo $\mathcal{A} \models T$
e definiamo quindi $\text{Mod}(T) = \{\mathcal{A} : \mathcal{A} \models T\}$
(nota: una teoria si dice *soddisfacibile* se ha un modello)
- **Conseguenza logica di una teoria** - E t.c. E è vero in tutti i modelli di T (diciamo che T *implica logicamente* E)
definiamo coerentemente $\text{Conseq}(T) = \{E : T \models E\}$

Mentre la teoria di una singola struttura è sempre completa, l'insieme delle sue conseguenze logiche non lo è necessariamente

Esempio: teoria dei gruppi

Un esempio è la teoria formata dagli assiomi di gruppo scritti in un linguaggio predicativo $\mathcal{L}_G = \mathcal{L}_{\text{Gruppi}}\{\circ, e\}$, dove \circ è un simbolo di funzione di arietà due ed e è un simbolo di costante.

Gli assiomi della teoria di gruppo si esprimono con i seguenti enunciati:

$$\begin{aligned} & \forall v_1 \forall v_2 \forall v_3 ((v_1 \circ v_2) \circ v_3 = v_1 \circ (v_2 \circ v_3)) \\ & \forall v_1 ((v_1 \circ e = v_1) \wedge (e \circ v_1 = v_1)) \\ & \forall v_1 \exists v_2 ((v_1 \circ v_2 = e) \wedge (v_2 \circ v_1 = e)) \end{aligned}$$

che formano la teoria dei gruppi T_G .

I modelli della teoria dei gruppi sono le strutture che chiamiamo gruppi, quindi si ha che $\text{Mod}(T_G) =$ insieme di tutti e soli i gruppi.

Gli enunciati che sono veri in qualunque gruppo formano l'insieme delle conseguenze logiche della teoria T_{Gruppi} , ossia $\{E : T_{\text{Gruppi}} \models E\}$. Le proprietà di un singolo gruppo G formano la “teoria di G ” $\{E : G \models E\}$.

Un esempio di come invece le conseguenze di una teoria non siano necessariamente complete ci viene dalle conseguenze degli assiomi di gruppo (ossia la teoria T_{Gruppi}): dato che esistono gruppi abeliani e gruppi non-abeliani e dato che la proprietà di essere abeliano è esprimibile nel linguaggio dei gruppi con un enunciato predicativo ($C = \forall v_1 \forall v_2 (v_1 \circ v_2 = v_2 \circ v_1)$), esistono modelli di T_{Gruppi} che soddisfano C e modelli di T_{Gruppi} che non soddisfano C (ossia soddisfano $\neg C$). Dunque né C né $\neg C$ sono in $\text{Conseq}(T_{\text{Gruppi}})$ (né l'essere abeliano né l'essere non abeliano sono conseguenze degli assiomi di gruppo).

La teoria T_{Gruppi} è dunque incompleta.

Relazioni tra teorie e coerenza

Siano T e S due teorie sullo stesso linguaggio tali che $T \subseteq S$.

affermazione	risp	motivo
Se $T \models A$, allora $S \models A$?	VERO	Monotonia
Se $A \models S$, allora $A \models T$?	VERO	Definizione di estensione (i modelli diminuiscono).
Se T è coerente, allora S è coerente?	FALSO	S potrebbe aggiungere assiomi che contraddicono T .
Se S è coerente, allora T è coerente?	VERO	Se esiste un modello per tutto S , quel modello va bene anche per il sottoinsieme T .
Se T è completa, S (propria) è coerente?	FALSO	T ha già deciso tutto - se S aggiunge qualcosa di nuovo (non derivabile da T), crea incoerenza.

Valgono le seguenti proprietà:

- monotonia - $T \models \varphi \implies S \models \varphi$

- “ereditarietà” dei modelli - ogni modello di S è anche modello di T (tutto T si trova dentro S)

$$\text{Mod}(S) \subseteq \text{Mod}(T)$$

- coerenza - se S è corente, allora anche T lo è
(se T è corente, non è detto che S lo sia)

3.4. Isomorfismi tra strutture

Se esiste un isomorfismo tra due strutture \mathcal{A} e \mathcal{B} , significa che queste sono “indistinguibili”, ovvero che hanno le stesse proprietà. È dunque vero che $\mathcal{A} \cong \mathcal{B} \implies \mathcal{A}$ e \mathcal{B} soddisfano gli stessi enunciati? La risposta è sì (nel linguaggio per cui sono isomorfe).

Thm. 15: Isomorfismo e soddisfacibilità

$\mathcal{A} \cong \mathcal{B} \implies \mathcal{A}$ e \mathcal{B} soddisfano le stesse formule.

Sia h l’isomorfismo. Si ha quindi $\forall F(x_1, \dots, x_n), \forall (a_1, \dots, a_n) \in A^n$

$$\mathcal{A} \models F(x_1, \dots, x_n)[a_1, \dots, a_n] \iff \mathcal{B} \models F(x_1, \dots, x_n)[h(a_1), \dots, h(a_n)]$$

Lo dimostriamo per le formule, che sono una struttura induttiva (così che valga anche per gli enunciati).

■ caso base:

$$(1) \ t_1 \leq t_2$$

$$\begin{aligned} \mathcal{A} \models (x \leq y)[a_1, a_2] &\iff \mathcal{B} \models (x \leq y)[h(a_1), h(a_2)] \\ (a_1, a_2) \in \leq_A &\iff (h(a_1), h(a_2)) \in \leq_B \\ &\text{vero per def. di isomorfismo} \end{aligned}$$

$$(2) \ t_1 = t_2$$

$$\begin{aligned} \mathcal{A} \models (x = y)[a_1, a_2] &\iff \mathcal{B} \models (x = y)[h(a_1), h(a_2)] \\ a_1 = a_2 &\iff h(a_1) = h(a_2) \\ &\text{vero per def. di isomorfismo} \end{aligned}$$

■ passo induttivo: Assumiamo che la proprietà valga per le formule ϕ e ψ .

Sia $\vec{a} = (a_1, \dots, a_n)$ la sequenza degli elementi in A e $h(\vec{a}) = (h(a_1), \dots, h(a_n))$ la sequenza in B .

(1) negazione (\neg):

$$\begin{aligned} \mathcal{A} \models \neg\phi[\vec{a}] &\\ \iff \mathcal{A} \not\models \phi[\vec{a}] &\quad (\text{def. di } \neg) \\ \iff \mathcal{B} \not\models \phi[h(\vec{a})] &\quad (\text{ip. ind. su } \phi) \\ \iff \mathcal{B} \models \neg\phi[h(\vec{a})] &\quad (\text{def. di } \neg) \end{aligned}$$

(2) congiunzione (\wedge):

$$\begin{aligned} \mathcal{A} \models (\phi \wedge \psi)[\vec{a}] &\\ \iff \mathcal{A} \models \phi[\vec{a}] \text{ e } \mathcal{A} \models \psi[\vec{a}] &\quad (\text{def. di } \wedge) \\ \iff \mathcal{B} \models \phi[h(\vec{a})] \text{ e } \mathcal{B} \models \psi[h(\vec{a})] &\quad (\text{ip. ind. su } \phi \text{ e } \psi) \\ \iff \mathcal{B} \models (\phi \wedge \psi)[h(\vec{a})] &\quad (\text{def. di } \wedge) \end{aligned}$$

(3) **quantificazione (\exists)**: Sia ϕ con variabile libera y in aggiunta a x_1, \dots, x_n .

$$\begin{aligned} \mathcal{A} \models (\exists y\phi)[\vec{a}] \\ \iff \exists c \in A \text{ tale che } \mathcal{A} \models \phi[\vec{a}, c] \quad (\text{def. di } \exists) \\ \iff \exists c \in A \text{ tale che } \mathcal{B} \models \phi[h(\vec{a}), h(c)] \quad (\text{ip. ind. su } \phi) \end{aligned}$$

Poiché h è suriettiva, l'insieme $\{h(c) \mid c \in A\}$ è uguale all'insieme base B . Posto $b = h(c)$:

$$\begin{aligned} \iff \exists b \in B \text{ tale che } \mathcal{B} \models \phi[h(\vec{a}), b] \\ \iff \mathcal{B} \models (\exists y\phi)[h(\vec{a})] \quad (\text{def. di } \exists) \end{aligned}$$

(Il caso per \forall segue da $\forall y\phi \equiv \neg\exists y\neg\phi$).

□

Passiamo ad un caso ancora più generale. Vogliamo mostrare che un isomorfismo tra due strutture

$$\begin{aligned} \mathcal{A} &= (A, \{R_i^A\}_{i \in I}, \{f_j^A\}_{j \in J}, \{c_k^A\}_{k \in K}) \\ \mathcal{B} &= (B, \{R_i^B\}_{i \in I}, \{f_j^B\}_{j \in J}, \{c_k^B\}_{k \in K}) \end{aligned}$$

preserva *l'intera struttura*.

Quindi

Def. 24: Isomorfismo tra due strutture

Siano \mathcal{A} e \mathcal{B} due strutture per il linguaggio \mathcal{L} .

\mathcal{A} è isomorfa a \mathcal{B} se e solo se esiste una mappa $h : \mathcal{A} \rightarrow \mathcal{B}$ biiettiva tale che:

(1) Per ogni simbolo di relazione n -ario R nel linguaggio, per ogni $(a_1, \dots, a_n) \in A^n$,

$$\mathcal{A} \models R(x_1, \dots, x_n)[a_1, \dots, a_n] \text{ se e solo se } \mathcal{B} \models R(x_1, \dots, x_n)[h(a_1), \dots, h(a_n)],$$

i.e.

$$(a_1, \dots, a_n) \in R^{\mathcal{A}} \iff (h(a_1), \dots, h(a_n)) \in R^{\mathcal{B}}.$$

(2) Per ogni simbolo di costante c nel linguaggio,

$$h(c^{\mathcal{A}}) = c^{\mathcal{B}},$$

(3) Per ogni simbolo di funzione n -ario f nel linguaggio, per ogni $(a_1, \dots, a_n) \in A^n$,

$$h(f^{\mathcal{A}}(a_1, \dots, a_n)) = f^{\mathcal{B}}(h(a_1), \dots, h(a_n)).$$

(h “commuta” con il simbolo di funzione f).

Vogliamo dimostrare che, per ogni termine t e ogni assegnazione α , il valore del termine t in \mathcal{A} con assegnazione α , quando mappato dall'isomorfismo h , è uguale al valore del termine t in \mathcal{B} con assegnazione $h(\alpha)$:

$$h(t^{\mathcal{A}, \alpha}) = t^{\mathcal{B}, h(\alpha)}$$

Si dimostra per induzione sulla struttura del termine t .

proof

(1) **Caso 1:** t è una variabile ($t = x$)

$$\begin{aligned} h(x^{\mathcal{A},\alpha}) &= h(\alpha(x)) && \text{(per def. di valutazione: } x^{\mathcal{A},\alpha} = \alpha(x)\text{)} \\ x^{\mathcal{B},h(\alpha)} &= h(\alpha)(x) && \text{(per def. di valutazione in } \mathcal{B} \text{ con assegnazione } h(\alpha)\text{)} \end{aligned}$$

Per definizione dell'assegnazione $h(\alpha)$, si ha $h(\alpha)(x) = h(\alpha(x))$. Quindi, l'uguaglianza è soddisfatta:

$$h(x^{\mathcal{A},\alpha}) = h(\alpha(x)) = h(\alpha)(x) = x^{\mathcal{B},h(\alpha)}$$

(2) **Caso 2:** t è una costante ($t = c$)

(in questo caso α non ha impatto - non ci sono assegnazioni)

$$\begin{aligned} h(c^{\mathcal{A},\alpha}) &= h(c^{\mathcal{A}}) && \text{(la valutazione di } c \text{ è l'interpretazione } c^{\mathcal{A}}\text{)} \\ &= c^{\mathcal{B}} && \text{(poiché } h \text{ è un isomorfismo, preserva le costanti)} \\ c^{\mathcal{B},h(\alpha)} &= c^{\mathcal{B}} && \text{(la valutazione di } c \text{ è l'interpretazione } c^{\mathcal{B}}\text{)} \end{aligned}$$

Quindi, l'uguaglianza è soddisfatta:

$$h(c^{\mathcal{A},\alpha}) = c^{\mathcal{B}} = c^{\mathcal{B},h(\alpha)}$$

(3) **Caso 3:** t è una funzione ($t = f(t_1, \dots, t_p)$)

$$\begin{aligned} h(t^{\mathcal{A},\alpha}) &= h(f(t_1, \dots, t_p)^{\mathcal{A},\alpha}) \\ &= h(f^{\mathcal{A}}(t_1^{\mathcal{A},\alpha}, \dots, t_p^{\mathcal{A},\alpha})) && \text{(def. di valutazione di termine funzione)} \\ &= f^{\mathcal{B}}(h(t_1^{\mathcal{A},\alpha}), \dots, h(t_p^{\mathcal{A},\alpha})) && \text{(\(h\) isomorfismo, preserva la funzione } f\text{)} \\ &= f^{\mathcal{B}}(t_1^{\mathcal{B},h(\alpha)}, \dots, t_p^{\mathcal{B},h(\alpha)}) \\ &= f(t_1, \dots, t_p)^{\mathcal{B},h(\alpha)} && \text{(def. di valutazione di termine funzione in } \mathcal{B}\text{)} \\ &= t^{\mathcal{B},h(\alpha)} \end{aligned}$$

La proprietà è dimostrata per tutti i termini. □

Corollary for Thm. 15: Equivalenza elementare

Si ha quindi che $\mathcal{A} \cong \mathcal{B} \implies \mathcal{A} \equiv \mathcal{B}$ (“elementarmente equivalenti”)

ovvero, \mathcal{A} e \mathcal{B} soddisfano gli stessi enunciati.

Thm. 16: Equivalenza elementare su domini finiti

In generale, non vale $\mathcal{A} \equiv \mathcal{B} \implies \mathcal{A} \cong \mathcal{B}$. Vale però se \mathcal{A} e \mathcal{B} finite e \mathcal{L} ha solo simboli di relazione.

Dimostrazione. Sia \mathcal{A} una struttura finita con dominio A di cardinalità n . Possiamo enumerare gli elementi del dominio come $A = \{a_1, a_2, \dots, a_n\}$.

L'idea della dimostrazione è costruire un singolo enunciato $\sigma_{\mathcal{A}}$ che descriva completamente la struttura \mathcal{A} a meno di isomorfismo.

Costruiamo $\sigma_{\mathcal{A}}$ come segue:

$$\sigma_{\mathcal{A}} = \exists x_1 \dots \exists x_n (\psi_{\text{card}} \wedge \psi_{\text{strut}})$$

1. Fissare la cardinalità La formula ψ_{card} assicura che il dominio abbia esattamente n elementi distinti:

$$\psi_{\text{card}} = \left(\bigwedge_{1 \leq i < j \leq n} x_i \neq x_j \right) \wedge \left(\forall y \bigvee_{i=1}^n y = x_i \right)$$

La prima parte afferma che gli x_i sono tutti diversi tra loro, mentre la seconda afferma che ogni elemento y del dominio coincide con uno degli x_i .

2. Fissare la struttura La formula ψ_{strut} descrive tutte le relazioni che valgono (e non valgono) tra gli elementi. Per ogni simbolo di relazione R in L (supponiamo WLOG che L contenga un solo simbolo di relazione binaria E):

$$\psi_{\text{strut}} = \left(\bigwedge_{(a_i, a_j) \in E^{\mathcal{A}}} E(a_i, a_j) \right) \wedge \left(\bigwedge_{(a_i, a_j) \notin E^{\mathcal{A}}} \neg E(a_i, a_j) \right)$$

Conclusione Chiaramente $\mathcal{A} \models \sigma_{\mathcal{A}}$, poiché basta interpretare le variabili x_i con gli elementi a_i . Dato che $\mathcal{A} \equiv \mathcal{B}$, allora anche $\mathcal{B} \models \sigma_{\mathcal{A}}$.

Ciò significa che esistono elementi b_1, \dots, b_n nel dominio di \mathcal{B} che soddisfano il corpo della formula $\sigma_{\mathcal{A}}$. Definiamo la mappa $f : A \rightarrow B$ ponendo $f(a_i) = b_i$.

Grazie a ψ_{card} , sappiamo che i b_i sono distinti e coprono tutto il dominio di \mathcal{B} , quindi f è una biiezione. Grazie a ψ_{strut} , la relazione è preservata:

$$(a_i, a_j) \in E^{\mathcal{A}} \iff \mathcal{B} \models E(b_i, b_j) \iff (f(a_i), f(a_j)) \in E^{\mathcal{B}}$$

Pertanto, f è un isomorfismo e $\mathcal{A} \cong \mathcal{B}$. □

3.5. La teoria DLO

3.5.1. Numeri razionali

Prendiamo la struttura $\mathcal{Q} = (\mathbb{Q}, \leq)$. Le proprietà di questa struttura si possono esprimere in enunciati predicativi nel linguaggio $\mathcal{L} = \{\leq(x, y)\}$, per il quale, per comodità, usiamo la notazione infissa $x \leq y$.

Aggiungiamo anche $x < y$ come abbreviazione di $(x \leq y) \wedge \neg(x = y)$.

Le proprietà di \mathcal{Q}, \leq sono le seguenti:

- (1) (A1 - Riflessività) $\forall x (x \leq x)$,
- (2) (A2 - Transitività) $\forall x \forall y \forall z ((x \leq y \wedge y \leq z) \rightarrow x \leq z)$,
- (3) (A3 - Antisimmetria) $\forall x \forall y ((x \leq y \wedge y \leq x) \rightarrow y = x)$,
- (4) (A4 - Totalità) $\forall x \forall y (x \leq y \vee x \leq y)$,
- (5) (A4 - Illimitato a destra) $\forall x \exists y (x < y)$,
- (6) (A5 - Illimitato a sinistra) $\forall x \exists y (y < x)$,
- (7) (A6 - Densità) $\forall x \forall y (x < y \rightarrow \exists z (x < z \wedge z < y))$.

Chiamiamo questa teoria **DLO** (*Dense Linear Order*).

Thm. 17: \mathbb{Q} e DLO

\mathbb{Q} è l'unico ordine lineare denso numerabile senza estremi a meno di isomorfismi.

3.5.1.1. Isomorfismo: caso numerabile

Thm. 18: Isomorfismo: caso numerabile

Sia $\mathcal{B} = (B, \leq_B)$ una struttura con dominio B numerabile e \leq_B un ordine totale denso senza estremi su B . Allora esiste un isomorfismo tra \mathcal{B} e \mathcal{Q} .

Vogliamo dimostrare che *ogni struttura* dove B è numerabile e \leq è un ordine totale denso senza estremi su A è isomorfa a (\mathbb{Q}, \leq) .

È naturale definire l'isomorfismo così: $\mathcal{B} \cong \mathcal{Q} \iff \exists h : \mathbb{Q} \rightarrow B$ t.c. $\forall q, q' \in \mathbb{Q}$

$$q \leq q' \iff h(q) \leq_B h(q')$$

$$q = q' \iff h(q) = h(q')$$

La dimostrazione usa un metodo chiamato *Back-and-Forth*.

proof

Vogliamo costruire un isomorfismo tra \mathcal{B} e \mathcal{Q} .

Entrambe hanno un dominio numerabile, quindi fissiamo una enumerazione di B (b_0, b_1, b_2, \dots) e una enumerazione di \mathbb{Q} (q_0, q_1, q_2, \dots).

Definiremo ricorsivamente una enumerazione p_1, p_2, p_3, \dots e una enumerazione d_1, d_2, d_3, \dots in modo tale che la mappa $p_i \mapsto d_i$ sia un isomorfismo tra \mathcal{B} e \mathcal{Q} .

Per induzione:

- Poniamo $p_0 = b_0$ e $d_0 = q_0$.
- Consideriamo un n generico e assumiamo che p_0, \dots, p_n e d_0, \dots, d_n siano definiti.

Distinguiamo due casi:

- (1) Caso **pari**: scegliamo un elemento p_{n+1} in $B \setminus \{p_0, \dots, p_n\}$ con indice minimo in (b_0, b_1, b_2, \dots) . Compariamo questo elemento agli elementi già scelti.

Abbiamo tre casi:

- (Prima di tutti gli altri elementi)

Per ogni $m \leq n$, $p_{n+1} < p_m$. In questo caso scelgo un q_{n+1} in \mathbb{Q} tale che per ogni $m \leq n$ abbiamo $d_{n+1} < d_m$. Questo elemento esiste perché \mathbb{Q} soddisfa gli assiomi che assicurano la *non esistenza di estremi* nell'ordine.

(Scelgo un elemento da \mathbb{Q} che sia anch'esso prima di tutti gli altri)

- (Dopo tutti gli altri elementi)

Per ogni $m \leq n$, $d_{n+1} > d_m$. In questo caso scelgo un q_{n+1} in \mathbb{Q} tale che per ogni $m \leq n$ valga $d_{n+1} > d_m$. Questo elemento esiste perché \mathbb{Q} soddisfa l'assioma che *esclude l'esistenza di un estremo destro* nell'ordine.

(Scelgo un elemento da \mathbb{Q} che sia anch'esso dopo tutti gli altri)

- (Da qualche parte nel mezzo)

Nessuno dei primi due casi. Allora esistono $m_0, m_1 \leq n$ tali che $p_{m_0} < p_{n+1} < p_{m_1}$ e nessun altro elemento di $\{p_0, p_1, \dots, p_n\}$ è nell'intervallo $[p_{m_0}, p_{m_1}]$. In questo caso scelgo un elemento d_{n+1} in \mathbb{Q} tale che $d_{m_0} < d_{n+1} < d_{m_1}$. Questo elemento esiste perché \mathbb{Q} soddisfa l'assioma di *densità*.

(Scelgo un elemento da \mathbb{Q} che sia anch'esso in mezzo agli altri)

- (2) Caso **dispari**: procediamo allo stesso modo, ma partendo da un elemento $q_{n+1} \in \mathbb{Q} \setminus \{q_0, \dots, q_n\}$.

Grazie alla separazione in passi pari e dispari (poiché scegliamo sia a partire da B che da \mathbb{Q}), non ci saranno “buchi” nelle nostre scelte di elementi.

Notiamo che nella dimostrazione non servono proprietà di \mathbb{Q} se non quelle di DLO (e l'ipotesi di numerabilità del dominio). Il teorema si può quindi generalizzare.

Thm. 19: Teorema di isomorfismo di Cantor

Siano $\mathcal{A} = (A, \leq_A)$ e $\mathcal{B} = (B, \leq_B)$ t.c. A, B sono entrambi numerabili e $\mathcal{A} \models \text{DLO}$ e $\mathcal{B} \models \text{DLO}$. Allora, $\mathcal{A} \cong \mathcal{B}$.

3.6. DLO: caso generale

Ci chiediamo: data una struttura $\mathcal{D} = (D, \leq_D) \models \text{DLO}$ con D più che numerabile, $\mathcal{D} \stackrel{?}{\equiv} \mathcal{Q}$ - ovvero, per esempio, esiste un enunciato nel linguaggio degli ordini vero in \mathbb{Q} e falso in \mathbb{R} ? (Se potessimo scrivere la completezza di \mathbb{R} in linguaggio degli ordini, allora questo sarebbe sicuramente vero (in quanto \mathbb{Q} non è completo) - ma non è possibile farlo).

Vogliamo quindi individuare un **criterio sufficiente** a concludere $\mathcal{Q} \equiv \mathcal{R}$.

3.6.1. Sottostrutture, proprietà del testimone

Introduciamo come prima cosa il concetto di sottostruttura:

Def. 25: Sottostruttura

Date due strutture \mathcal{A} e \mathcal{B} , \mathcal{A} si dice **sottostruttura** di \mathcal{B} ($\mathcal{A} \subseteq \mathcal{B}$) se e solo se si ha:

- $A \subseteq B$
- per ogni simbolo R di relazione, $R^{\mathcal{A}} = R^{\mathcal{B}} \cap A^n$ (dove n arità di R)
- per ogni simbolo di costante, $c^{\mathcal{B}} \in A$ e $c^{\mathcal{A}} = c^{\mathcal{B}}$
- per ogni simbolo di funzione, $f^{\mathcal{A}} = f^{\mathcal{B}}|_A^n$ (con n arità di f)

Def. 26: Sottostruttura elementare

Date due strutture \mathcal{A} e \mathcal{B} , \mathcal{A} si dice **sottostruttura elementare** di \mathcal{B} ($\mathcal{A} \prec \mathcal{B}$) se e solo se si ha:

$$A \subseteq B \wedge \forall F, \forall \vec{a} \in A, \mathcal{A} \models F[\vec{a}] \iff \mathcal{B} \models F[\vec{a}]$$

(ovvero $\mathcal{A} \subseteq \mathcal{B}$ e $\mathcal{A} \equiv \mathcal{B}$).

Notiamo che $\mathcal{Q} \subseteq \mathcal{R}$.

Data una formula F , è naturale che se α è un assegnamento in \mathcal{Q} e $\mathcal{Q} \models \exists x F[\alpha]$, allora $\mathcal{R} \models \exists x F[\alpha]$.

Non è però detto che valga l'inverso, ossia:

Thm. 20: Proprietà del testimone tra \mathcal{R} e \mathcal{Q}

Per ogni formula F con x variabile libera, *per ogni assegnamento* α in \mathcal{Q} , se

$$\mathcal{R} \models \exists x F(x)[\alpha]$$

allora $\exists q \in \mathbb{Q}$ tale che:

$$\mathcal{R} \models F(x)[\alpha(\frac{x}{q})]$$

(quindi, abbiamo una formula F con tutte le variabili in \mathcal{Q} tranne una; presumiamo che sia vera con l'ultima variabile assegnata in \mathbb{R} - allora vogliamo che $\exists q \in \mathbb{Q}$ (da assegnare all'ultima variabile) che la soddisfi)

La proprietà del testimone è la condizione sufficiente per dedurre $\mathcal{Q} \equiv \mathcal{R}$.

Come già visto, conviene dimostrare la proprietà per le formule (così che valga anche per gli enunciati).

Corollary for Thm. 20: Conseguenza della proprietà del testimone

Se vale la proprietà del testimone tra \mathcal{R} e \mathcal{Q} , allora $\forall F, \forall \alpha$ su \mathcal{Q} ,

$$\mathcal{Q} \models F[\alpha] \iff \mathcal{R} \models F[\alpha]$$

proof

Si dimostra per induzione su F .

■ **caso base:** F è di tipo $(x \leq y)$.

$$\begin{aligned}\mathcal{Q} \models (x \leq y)[\alpha] &\iff \mathcal{R} \models (x \leq y)[\alpha] \\ \alpha(x) \leq_{\mathbb{Q}} \alpha(y) &\iff \alpha(x) \leq_{\mathbb{R}} \alpha(y)\end{aligned}$$

(argomento analogo per $(x = y)$)

■ **booleani:**

$$\begin{aligned}\mathcal{Q} \models A \wedge B[\alpha] &\iff \mathcal{R} \models A \wedge B[\alpha] \\ \mathcal{Q} \models A[\alpha] \wedge \mathcal{Q} \models B[\alpha] &\iff \mathcal{R} \models A[\alpha] \wedge \mathcal{R} \models B[\alpha]\end{aligned}$$

Vero per ipotesi induttiva su A e B :

$$\mathcal{Q} \models A[\alpha] \iff \mathcal{R} \models A[\alpha] \quad \text{e} \quad \mathcal{Q} \models B[\alpha] \iff \mathcal{R} \models B[\alpha]$$

(argomento analogo per gli altri connettivi booleani)

■ **esistenziale:**

Sia $F \equiv \exists x F(x)$. Assumiamo che la tesi valga per la formula $F(x)$ (I.I.). Dobbiamo mostrare che:

$$\mathcal{Q} \models \exists x F(x)[\alpha] \iff \mathcal{R} \models \exists x F(x)[\alpha]$$

Direzione \implies : Assumiamo $\mathcal{Q} \models \exists x F(x)[\alpha]$. Per definizione, vale sse esiste un elemento $q \in \mathbb{Q}$ tale che $\mathcal{Q} \models F(x)[\alpha(\frac{x}{q})]$. Poiché l'assegnazione $\alpha(\frac{x}{q})$ è in \mathbb{Q} , per I.I. applicata a $F(x)$:

$$\mathcal{Q} \models F(x)[\alpha(\frac{x}{q})] \iff \mathcal{R} \models F(x)[\alpha(\frac{x}{q})]$$

Poiché $q \in \mathbb{Q}$ e $\mathbb{Q} \subseteq \mathbb{R}$, q è un elemento di \mathbb{R} . Per definizione di soddisfacibilità in \mathbb{R} :

$$\mathcal{R} \models \exists x F(x)[\alpha]$$

Direzione \impliedby : Assumiamo $\mathcal{R} \models \exists x F(x)[\alpha]$, con α assegnamento in \mathbb{Q} . Per definizione, vale sse $\exists r \in \mathbb{R}$ t.c. $\mathcal{R} \models F(x)(\frac{x}{r})$. Per la proprietà del testimone, se $\mathcal{R} \models \exists x F(x)[\alpha]$, allora esiste un testimone $q \in \mathbb{Q}$ tale che $\mathcal{R} \models F(x)[\alpha(\frac{x}{q})]$. Per ipotesi induttiva su F , questo vale sse $\exists q \in \mathbb{Q}$ t.c. $\mathcal{Q} \models F(x)[\alpha(\frac{x}{q})]$. Ovvero:

$$\mathcal{R} \models F(x)[\alpha(\frac{x}{q})] \iff \mathcal{Q} \models F(x)[\alpha(\frac{x}{q})]$$

Quindi $\mathcal{Q} \models F(x)[\alpha(\frac{x}{q})]$. Poiché esiste un $q \in \mathbb{Q}$ che soddisfa $F(x)$ in \mathbb{Q} , per definizione (di soddisfacibilità):

$$\mathcal{Q} \models \exists x F(x)[\alpha]$$

□

Corollary for Thm. 20: Altra conseguenza della proprietà del testimone

Quindi, se vale la proprietà del testimone tra \mathcal{Q} e \mathcal{R} , per ogni enunciato E nel linguaggio degli ordini si ha:

$$\mathcal{Q} \models E \iff \mathcal{R} \models E$$

ovvero $\mathcal{Q} \equiv \mathcal{R}$, e quindi $\mathcal{Q} \prec \mathcal{R}$.

3.6.2. Equivalenza tra \mathcal{R} e \mathcal{Q}

Per dimostrare $\mathcal{Q} \equiv \mathcal{R}$, definiamo $\mathcal{A} = (A, \leq_A)$ t.c. $\mathcal{Q} \subseteq \mathcal{A} \subseteq \mathcal{R}$ e A numerabile.

Seguiremo quindi questi passi:

- (1) dimostreremo la proprietà del testimone tra \mathcal{A} e \mathcal{R} , ottenendo $\mathcal{A} \equiv \mathcal{R}$
- (2) visto che $\mathcal{R} \models \text{DLO}$ e (1), si ha $\mathcal{A} \models \text{DLO}$
- (3) da (2) e A numerabile, applicando il Teorema di Cantor (p.41) si ottiene $\mathcal{A} \cong \mathcal{Q}$
- (4) visto che $\cong \implies \equiv$, si ha $\mathcal{A} \equiv \mathcal{Q}$
- (5) per transitività, $\mathcal{R} \equiv \mathcal{Q}$.

Costruiamo quindi \mathcal{A} partendo da \mathcal{Q} .

Sia F una formula del linguaggio DLO e siano x_1, x_2, y le sue variabili libere.

Se $\mathcal{R} \models \exists y F(x_1, x_2, y) \begin{bmatrix} x_1 & x_2 \\ q_1 & q_2 \end{bmatrix}$, per definizione di soddisfacibilità sappiamo che $\exists r \in \mathbb{R}$ t.c. $\mathcal{R} \models F(x_1, x_2, y) \begin{bmatrix} x_1 & x_2 & y \\ q_1 & q_2 & r \end{bmatrix}$. Ne scegliamo uno in modo canonico (usiamo implicitamente l'assioma della scelta).

Def. 27: Funzione di Skolem

Chiamiamo **funzione di Skolem** (da Thoralf Skolem) della formula F relativamente ad y la funzione

$$f_{F,y} : \mathcal{Q}^n \rightarrow \mathcal{R}$$

che associa ad ogni scelta di (q_1, \dots, q_n) un tale $r \in \mathbb{R}$ in modo canonico.

Dato $\mathcal{F} = \{\text{funzioni di Skolem}\}$, chiudiamo \mathcal{Q} sotto \mathcal{F} .

Otteniamo $A_1 = \mathcal{Q} \cup \{b \in \mathbb{R} \mid b = f(q_1, \dots, q_n) \text{ con } f \in \mathcal{F}, (q_1, \dots, q_n) \in \mathcal{Q}^n\}$.

Sappiamo che $\mathcal{Q} \subseteq \mathcal{A}_1 = (A_1, \leq_A) \subseteq \mathcal{R}$.

È però vero che se x_1, x_2 sono assegnati in A_1 come a_1, a_2 allora esiste un $a \in A_1$ t.c.

$\mathcal{R} \models F(x_1, x_2, y) \begin{bmatrix} x_1 & x_2 & y \\ a_1 & a_2 & a \end{bmatrix}$? Non necessariamente. Infatti, se $a_1, a_2 \in \mathcal{Q}$, sicuramente sì.

Ma, se $a_1, a_2 \in A_1 \setminus \mathcal{Q}$, non possiamo saperlo per certo. Dobbiamo quindi chiudere nuovamente sotto \mathcal{F} .

Non ci basta però chiudere un'altra volta, in quanto il problema si ripropone. Dobbiamo ripetere l'operazione $\forall i \geq 0$.

Abbiamo quindi

$$A_{i+1} = A_i \cup \mathcal{F}(A_i)$$

(utilizziamo la notazione $\mathcal{F}(A)$ per indicare la chiusura di A su F)

Definiamo

$$A = \bigcup_{i \in \mathbb{N}} A_i$$

Si vede che A è chiuso sotto funzione di Skolem.

Infatti, se $a_1, \dots, a_n \in A_k$ e $f \in \mathcal{F}$ allora $f(a_1, \dots, a_n) \in A_{k+1}$.

$\mathcal{A} = (A = \bigcup_{i \in \mathbb{N}} A_i, \leq_A)$ ha anche la Proprietà del Testimone relativamente ad \mathcal{R} .

Ovvero, se

$$\mathcal{R} \models \exists y F(x_1, x_2, y) \begin{bmatrix} x_1 & x_2 \\ a_1 & a_2 \end{bmatrix} \quad a_1, a_2 \in A$$

allora $\exists a \in A$ t.c.

$$\mathcal{R} \models F(x_1, x_2, y) \begin{bmatrix} x_1 & x_2 & y \\ a_1 & a_2 & a \end{bmatrix}$$

[caso generale: $\forall F, \forall \alpha \in A$, se:

$$\mathcal{R} \models \exists x F(x)[a]$$

allora $\exists a \in A$ t.c.:

$$\mathcal{A} \models F(x) [\alpha(\frac{x}{a})]$$

]

Possiamo quindi concludere che $\mathcal{A} \equiv \mathcal{R}$.

Possiamo anche verificare che \mathcal{A} è numerabile:

- \mathcal{F} è numerabile (perché esiste una funzione per ogni scelta di formula del linguaggio e di variabile libera, entrambe da insiemi numerabili)
- ogni A_k è numerabile ($A_1 = \mathbb{Q}$ è numerabile, e ogni A_{k+1} è l'unione di un A_k numerabile con la sua chiusura sotto un altro insieme numerabile)

Abbiamo quindi dimostrato la seguente proposizione

Prop. 1

Esiste una sottostruttura numerabile \mathcal{A} di \mathcal{R} che contiene \mathbb{Q} e soddisfa esattamente gli stessi enunciati di \mathcal{R} nel linguaggio degli ordini.

\mathcal{A} è anche un modello numerabile di DLO, perché $\mathcal{R} \models \text{DLO}$.

Dunque $\mathcal{A} \cong \mathcal{Q}$. Dunque $\mathcal{Q} \equiv \mathcal{R}$ (nel linguaggio degli ordini).

3.6.3. Generalizzazione della dimostrazione

La dimostrazione è generale - si può ricostruire in maniera equivalente partendo non da \mathcal{R} ma da un qualunque \mathcal{B} modello non-numerabile di DLO.

La Proprietà del Testimone si esprime non relativamente a \mathcal{Q} ma ad una sottostruttura numerabile \mathcal{X} di \mathcal{B} del tipo $(X, \leq^{\mathcal{X}})$ con $X \subseteq B$ e $\leq^{\mathcal{A}} = \leq^{\mathcal{B}} \cap (A \times A)$.

Si parte da un sottoinsieme numerabile X di \mathcal{B} (che ha sicuramente), e si procede per chiusura sotto funzioni di Skolem allo stesso modo.

Si costruisce così $\mathcal{X} = \bigcup_{i \in \mathbb{N}} X_i$. Si ha $\mathcal{X} \models \text{DLO}$, e quindi $\cong \equiv \mathcal{Q}$.

Thm. 21

Si ha quindi che **tutti i modelli di DLO** (numerabili o meno) soddisfano esattamente gli stessi enunciati di \mathcal{Q} .

i.e. $\mathcal{B} \models \text{DLO} \implies \mathcal{B} \equiv \mathcal{Q}$

DLO “identifica” quindi $(\mathcal{Q}, \leq) : Th(\mathcal{Q}) = Th(\mathcal{B})$

Corollary for Thm. 21:

DLO è una teoria completa.

(ossia, \forall enunciato E , DLO $\models E$ oppure DLO $\models \neg E$ (e DLO ha almeno un modello)).

Un modello di DLO soddisfa esattamente gli stessi enunciati di \mathcal{Q} . Dunque, $Cons(DLO)$ sono esattamente gli enunciati veri nella singola struttura di \mathcal{Q} , che è una teoria completa.

Vademecum per esercizi su DLO

Riassumiamo alcune informazioni utili su DLO.

Poiché DLO è completa e \mathcal{Q} ne è un modello, e poiché DLO e \mathbb{R} condividono la stessa teoria nel linguaggio degli ordini, vale:

$$DLO \models E \iff \mathcal{Q} \models E \iff \mathcal{R} \models E$$

Questo significa che (limitatamente al linguaggio degli ordini)

- se un enunciato E è vero in \mathbb{R} , DLO $\models E$
- se un enunciato E è falso in \mathbb{Q} , DLO $\models \neg E$

esempio: se sappiamo che $\mathcal{Q} \models (E \vee F) \wedge \neg(E \wedge F)$ e che $\mathcal{R} \models F$, è necessariamente vero che DLO $\models \neg E$?

- abbiamo che $\mathcal{R} \models F \implies \mathcal{Q} \models F$
- la formula $(E \vee F) \wedge \neg(E \wedge F)$ è essenzialmente uno XOR su E ed F
- poiché $\mathcal{Q} \models F$, otteniamo che $\mathcal{Q} \models \neg E$
- quindi, DLO $\models \neg E$

3.7. Criterio di Tarski-Vaught

Per stabilire se una sottostruttura è anche una **sottostruttura elementare** ($\mathcal{A} \preceq \mathcal{B}$), possiamo quindi utilizzare la *Proprietà del Testimone*, che chiamiamo anche “criterio di Tarski-Vaught”.

Def. 28: Proprietà del Testimone (generale)

Sia $\mathcal{A} \subseteq \mathcal{B}$. Diciamo che \mathcal{A} soddisfa la **Proprietà del Testimone** rispetto a \mathcal{B} se, per ogni formula del tipo $\exists x F(x)$ e per ogni assegnamento α in \mathcal{A} , se

$$\mathcal{B} \models \exists x F(x)[\alpha]$$

allora $\exists a \in A$ tale che:

$$\mathcal{B} \models F(x)[\alpha(a)]$$

(se in \mathcal{B} esiste una soluzione (un testimone) per una certa proprietà parametrizzata da elementi di A , dobbiamo essere in grado di trovare quel testimone già dentro A)

Thm. 22

Sia \mathcal{A} è una sottostruttura di \mathcal{B} che **soddisfa la Proprietà del Testimone** rispetto a \mathcal{B} , allora \mathcal{A} è una **sottostruttura elementare** di \mathcal{B} .

$$\mathcal{A} \subseteq \mathcal{B} \text{ e Testimone} \implies \mathcal{A} \preceq \mathcal{B} (\implies \mathcal{A} \equiv \mathcal{B})$$

Quindi, iniziamo a vedere che partendo da un arbitrario modello \mathcal{B} infinito, ne esiste sempre uno numerabile (purché il linguaggio sia numerabile).

Prendiamo $\mathcal{B} \models T$ con $|\mathcal{B}| = \infty$ e $X \subseteq B$ con X numerabile.

Facciamo la solita chiusura per funzioni di Skolem.

Immaginiamo una successione di insiemi $X_0 \subseteq X_1 \subseteq X_2 \dots$

Definiamo:

$$A = \bigcup_{i \in \mathbb{N}} X_i$$

Definiamo la struttura \mathcal{A} in questo modo:

- Per ogni simbolo di relazione: $R^{\mathcal{A}} = R^{\mathcal{B}} \cap A^n$
- Per ogni funzione: $f^{\mathcal{A}} = f^{\mathcal{B}}|_{A^n}$

NB: Ci manca la proprietà sulle costanti. Dobbiamo avere $c^{\mathcal{B}} \in A$. Possiamo trattare le costanti come funzioni di Skolem 0-arie (“casi base” della definizione induttiva), garantendo la loro inclusione nella chiusura.

La struttura risultante è:

$$\mathcal{A} = \left(\bigcup_{i \in \mathbb{N}} X_i, \{c_i^{\mathcal{A}}\}, \{R_j^{\mathcal{A}}\}, \{f_k^{\mathcal{A}}\} \right)$$

\mathcal{A} è una sottostruttura per definizione e ha la proprietà del testimone (Tarski-Vaught).

$$\implies \mathcal{A} \equiv \mathcal{B} \quad (\text{elementarmente equivalenti})$$

Quindi, $\mathcal{A} \models T$.

3.8. Teorema di Löwenheim-Skolem (All'in giù)

Thm. 23: Teorema di Löwenheim-Skolem all'in giù

Sia \mathcal{B} una struttura infinita adeguata per un linguaggio numerabile \mathcal{L} . Sia $X \subseteq B$ un sottoinsieme del suo dominio. Esiste una struttura \mathcal{A} tale che:

- (1) $X \subseteq A$
- (2) $\mathcal{A} \preceq \mathcal{B}$ (è sottostruttura elementare)
- (3) Se il linguaggio \mathcal{L} e l'insieme X sono numerabili, allora A è numerabile.

Essenzialmente, **ogni teoria in un linguaggio numerabile che ammette un modello infinito (\mathcal{B}), ammette un “sotto-modello” numerabile (\mathcal{A})** che è una sottostruttura elementare di \mathcal{B} .

La dimostrazione procede quasi analogamente a quella vista sopra.

Corollary for Thm. 23:

Non esiste una teoria T in un linguaggio \mathcal{L} numerabile (predicativo) che possa forzare i suoi modelli ad essere “più che numerabili”.

$$\mathcal{B} \models T \implies |B| > \aleph_0 \quad (\text{Falso in generale})$$

Corollary for Thm. 23: “Paradosso” di Skolem

La teoria assiomatica degli insiemi (ZFC) (se ha un modello) ha modelli numerabili.

(Il linguaggio ha un solo simbolo di relazione binaria (\in) ed è numerabile).

Il “paradosso” (non un vero paradosso) nasce dal fatto che:

- In ZFC si dimostra l'esistenza di insiemi non numerabili (es. l'insieme dei reali \mathbb{R}). Formalmente:

$$\text{ZFC} \vdash \exists x(\text{"}x \text{ è non numerabile"}\text{")}$$

- Se ZFC ha un modello \mathcal{V} , allora per il Teorema di Löwenheim-Skolem deve averne anche un modello numerabile \mathcal{M} .
- Quindi esiste una struttura numerabile in cui è soddisfatto l'enunciato “esiste un insieme non-numerabile”.

3.9. Teorie categoriche (ω -categoricità)

L'argomento usato per dimostrare la completezza di DLO si può generalizzare.

Def. 29: ω -categoricità)

Una teoria T si dice **ω -categorica** se tutti i suoi modelli **numerabili** sono isomorfi tra loro.

(ricordiamo che se due modelli sono isomorfi, allora sono necessariamente elementarmente equivalenti)

Sia T una teoria in un linguaggio numerabile che soddisfa le seguenti tre proprietà:

- (1) T è **soddisfacibile** (ha almeno un modello).
- (2) T **non ha modelli finiti** (ha solo modelli infiniti).
- (3) T è **ω -categorica** (ha un solo modello numerabile a meno di isomorfismo).

Possiamo concludere che T è una teoria **completa**.

proof

Supponiamo che T non sia completa.

1. Se T non è completa, esiste un enunciato E nel linguaggio tale che T non dimostra né E né $\neg E$.

$$T \not\models E \quad \text{e} \quad T \not\models \neg E$$

2. Questo implica che esistono due modelli \mathcal{B} e \mathcal{C} tali che:

$$\mathcal{B} \models T \cup \{\neg E\} \quad \text{e} \quad \mathcal{C} \models T \cup \{E\}$$

3. Per l'ipotesi (2), T non ha modelli finiti, quindi \mathcal{B} e \mathcal{C} sono entrambi infiniti.

4. Applichiamo il Teorema di Löwenheim-Skolem a entrambi i modelli:

- Esiste $\mathcal{B}_0 \preceq \mathcal{B}$ con \mathcal{B}_0 numerabile. Poiché $\mathcal{B} \models \neg E$, allora $\mathcal{B}_0 \models \neg E$.
- Esiste $\mathcal{C}_0 \preceq \mathcal{C}$ con \mathcal{C}_0 numerabile. Poiché $\mathcal{C} \models E$, allora $\mathcal{C}_0 \models E$.

5. Ora abbiamo due modelli numerabili di T , \mathcal{B}_0 e \mathcal{C}_0 . Per l'ipotesi (3) di **ω -categoricità**, tutti i modelli numerabili sono isomorfi:

$$\mathcal{B}_0 \cong \mathcal{C}_0$$

6. Se due strutture sono isomorfe, soddisfano gli stessi enunciati:

$$\mathcal{B}_0 \equiv \mathcal{C}_0$$

Tuttavia, abbiamo stabilito al punto 4 che $\mathcal{B}_0 \models \neg E$ e $\mathcal{C}_0 \models E$ (contraddizione) □

4

Calcolo dei Predicati

Introduciamo ora il calcolo dei predicati (con identità “alla Hilbert”). Ad ogni linguaggio \mathcal{L} possiamo associare un calcolo dei predicati del I ordine.

Def. 30: Dimostrabilità

Si ha $T \vdash A$ se e solo se esiste una sequenza (A_1, A_2, \dots, A_n) di formule tali che:

- $A_n = A$
- $\forall 1 \leq i \leq n:$
 - $A_i \in T$, oppure
 - A_i assioma, oppure
 - A_i segue dai precedenti per regole di inferenza

Gli assiomi logici sono i seguenti:

- (1) Tutti gli assiomi proposizionali (illustrati a p.25)

Gli assiomi predicativi:

- (2) $\forall x F \rightarrow F[x/t]$ con t termine libero per x in F (per “libero” vedi p.32)

da questa regola deriva anche:
$$\frac{F[x/t]}{\exists x F}$$

- (3) $\forall x(F \rightarrow G) \rightarrow (F \rightarrow \forall x G)$, con F senza occorrenze libere di x (si dice “ F non parla di x ”)

Gli assiomi dell’identità:

- (4) Per ogni simbolo di relazione R :

$$\forall x_1 \dots x_n, \forall y_1, \dots y_n, \left(\bigwedge_{i=1}^n (x_i = y_i) \right) \rightarrow (R(x_1, \dots, x_n) \leftrightarrow R(y_1, \dots, y_n))$$

- (5) Per ogni simbolo di funzione f :

$$\forall x_1 \dots x_n, \forall y_1, \dots y_n, \left(\bigwedge_{i=1}^n (x_i = y_i) \right) \rightarrow (f(x_1, \dots, x_n) = f(y_1, \dots, y_n))$$

(6) (uguaglianza e transitività)

$$\forall x y z \left((x = x) \wedge ((x = y) \rightarrow (y = x)) \wedge (((x = y) \wedge (y = z)) \rightarrow (x = z)) \right)$$

(valgono $(x = x)$ e $(x = y \rightarrow y = x)$ e $((x = y \wedge y = z) \rightarrow x = z)$)

Le **regole di inferenza** che utilizziamo nel calcolo predicativo sono:

$$(1) \text{ Modus Ponens: } \frac{X \quad X \rightarrow Y}{Y}$$

$$(2) \text{ Generalizzazione: } \frac{F}{\forall x F}$$

(attenzione! non è come dire che se una formula F vale in un caso, allora vale $\forall x$ - stiamo dicendo che, se ho dimostrato in maniera generica F , allora essa è vera $\forall x$)

4.0.1. Proprietà fondamentali del Calcolo dei Predicati

Il calcolo dei predicati mantiene molte delle proprietà del calcolo proposizionale, come:

- il **teorema di correttezza**: $T \vdash A \implies T \vDash A$ (anche qui tutti gli assiomi sono verità logiche e le regole di inferenza preservano le verità logiche)
- $T \vdash A \wedge T \subseteq S \implies S \vdash A$ (monotonia)
- se $\vdash A$, A è detto “teorema”
- $$\frac{T \vdash A \quad A \vdash B}{T \vdash B}$$
- $$\frac{T \vdash A \quad S \vdash B}{T \cup S \vdash A \wedge B}$$

Thm. 24: Teorema di deduzione per il calcolo predicativo

Sia E un enunciato, A una formula e Γ un insieme di formule. Vale:

$$\Gamma, E \vdash A \iff \Gamma \vdash (E \rightarrow A)$$

Altre proprietà:

- Ogni istanza di tautologia proposizionale è una verità nel Calcolo dei Predicati

4.0.2. Teorema di completezza per la logica predicativa

Come per la logica proposizionale, vogliamo dimostrare $T \vDash A \iff T \vdash A$.

Esiste però un'altra definizione equivalente per il teorema di completezza.

Def. 31: Coerenza

Una teoria T è **coerente** (o non-contraddittoria) $\iff \neg \exists$ enunciato A t.c. $T \vdash A \wedge T \vdash \neg A$

Il Calcolo dei Predicati è coerente, ossia per nessuna formula vale $\vdash F \wedge \vdash \neg F$.

Def. 32: Teorema di completezza v.2

$$T \text{ coerente} \iff T \text{ ha un modello } (\in \text{SAT})$$

Notiamo facilmente che T incoerente come testimoniato da A (ovvero $T \vdash A \wedge T \vdash \neg A$) implica necessariamente T insoddisfacibile.

Infatti, $T \vdash A \implies T \models A$ e $T \vdash \neg A \implies T \models \neg A$, e se valgono sia $T \models A$ che $T \models \neg A$ sappiamo che T è insoddisfacibile ($\text{Mod}(T) = \emptyset$).

Ci manca da dimostrare T coerente $\implies \text{Mod}(T) \neq \emptyset$.

Thm. 25: Equivalenza tra le due forme di teorema di completezza

$$\text{Si ha che } T \vdash A \iff T \models A_{(1)} \equiv T \text{ coerente} \iff T \text{ ha un modello } (\in \text{SAT})_{(2)}$$

proof

(1) Assumiamo (2) e dimostriamo $T \vdash A \implies T \models A$

Sappiamo che $T \models A$. Ci sono due opzioni:

- $T \in \text{UNSAT}$: $T \models A$ è vera $\forall A$, quindi T incoerente
(se $\exists B$ t.c. $T \vdash B \wedge T \vdash \neg B$, possiamo dimostrare una qualsiasi affermazione A in questo modo: $\vdash B \rightarrow (\neg B \rightarrow A)$)
quindi $T \vdash A$
- $T \in \text{SAT}$: se $T \in \text{SAT}$ e $T \models A$, sappiamo che $T \cup \neg A \in \text{UNSAT}$.
Per (2), quindi, $T \cup \neg A$ è incoerente, quindi $T \vdash A$
 $T \cup \neg A$ incoerente significa $T \cup \neg A \vdash \perp$, quindi (x deduzione) $T \vdash \neg A \rightarrow \perp$;
per reductio ad absurdum ($\neg A \rightarrow \perp \vdash A$), si ha quindi $T \vdash A$.

(2) Assumiamo (1) e dimostriamo T coerente $\implies T$ ha un modello

Presumiamo che T non abbia un modello. Per definizione di conseguenza logica, abbiamo $T \models \perp$.
Per (1), otteniamo $T \vdash \perp$, ovvero T incoerente.

4.0.3. Estensioni di teorie

Vogliamo dimostrare coerenza \implies soddisficiabilità. Per farlo, introduciamo il concetto di *estensione*.

Def. 33: Estensione

Diciamo che una teoria T' **estende** una teoria T se $T \subseteq T'$.

Def. 34: Teoria sintatticamente completa

Una teoria T si dice **sintatticamente completa** se per ogni enunciato E nel linguaggio di T , vale $T \vdash E$ oppure $T \vdash \neg E$.

Lemma 6: Lemma di Lindenbaum

Ogni teoria coerente (in un linguaggio numerabile) ammette un'estensione coerente e completa.

(Sia T coerente. Allora $\exists S$ teoria nel linguaggio di T t.c.:

- $T \subseteq S$
- S è coerente
- S è sintatticamente completa

)

proof

Fissiamo un'enumerazione $\{E_1, E_2, \dots\}$ di tutti gli enunciati di T .

Definiamo una successione di teorie in questo modo:

- $S_0 = T$
- $S_{n+1} = \begin{cases} S_n \cup \{E_{n+1}\} & \text{se } S_n \cup \{E_{n+1}\} \text{ coerente (ovvero } S_n \not\vdash \neg E_{n+1}) \\ S_n & \text{altrimenti (ovvero } S_n \vdash \neg E_{n+1}) \end{cases}$

nel secondo caso non serve aggiungere $\{\neg E_{n+1}\}$ perché si ha già $S_n \vdash \neg E_{n+1}$ (sarebbe superfluo)

Sia $S = \bigcup_{n \in \mathbb{N}} S_n$.

(1) S è coerente

S è coerente significa che $\nexists E$ per cui $\exists \{A_1, A_2, \dots, A_t\}, \{B_1, B_2, \dots, B_t\} \subseteq S$ t.c.

$A_1, A_2, \dots, A_t \vdash E$ e $B_1, B_2, \dots, B_t \vdash \neg E$.

Si nota facilmente che questo è impossibile, in quanto, per costruzione, entrambi gli insiemi di enunciati apparterrebbero a un S_m per qualche m (vista la costruzione “a catena” di S), e si avrebbe $S_m \vdash E$ e $S_m \vdash \neg E$, quindi S_m incoerente (ma per costruzione, $\forall i S_i$ è coerente).

(2) S è sintatticamente completa

Scegliamo di dimostrarlo nella forma $S \not\vdash \neg E \implies S \vdash E$

Sia $E = E_{n+1}$ per un qualche n . Sappiamo che $\forall i$ e in particolare per n , $S_n \not\vdash \neg E_{n+1}$.

Quindi, per costruzione $S_{n+1} = S_n \cup \{E_{n+1}\} \implies S_{n+1} \vdash E_{n+1} \xrightarrow{S_{n+1} \subseteq S} S \vdash E_{n+1}$.

Se invece di T coerente avessimo ipotizzato T soddisfacibile, la conclusione sarebbe stata banale perché si sarebbe definita S facilmente come (dato \mathcal{A} modello di T) $S = Th(\mathcal{A}) = \{E \mid \mathcal{A} \models E\}$, evidentemente coerente e completo.

Il teorema di completezza mostra che le due cose sono equivalenti.

Abbiamo quindi dimostrato che T coerente $\implies \exists S \supseteq T$ coerente e completa.

Vogliamo però T coerente $\implies T$ soddisfacibile.

Il passo successivo è dimostrare che S è già quasi un modello.

Cerchiamo quindi un modello $\mathcal{M} = (M, \mathcal{R}^{\mathcal{M}}, f_j^{\mathcal{M}}, c_i^{\mathcal{M}})$

Per definirlo, facciamo alcune assunzioni.

Assumiamo di lavorare con un linguaggio $\mathcal{L} = \{R_i, f_j, c_k\}$ che contenga almeno una costante e una funzione (nota bene: se ci sono una costante e una funzione, ci sono infiniti termini chiusi).

Definiamo \mathcal{M} (detto **modello di Henkin** dei termini di S):

- $M = \{\text{termini chiusi di } \mathcal{L}\}$
es. $\mathcal{L}_1 = \{0, 1, +\}, M = \{0, 1, 0+0, 0+1, 1+0, \dots\}$
- $c_k^M = c_k \in M$
l'interpretazione della costante c è data dalla costante c stessa
es. $\mathcal{L}_2 = \{e, *\}, M_2 = \{e, e * e, \dots\}$ - come interpretazione di $c_0 = e$ prendo $e \in M_2$
- l'interpretazione di un simbolo di relazione \mathcal{R} è data dall'insieme dei termini chiusi di cui S dimostra che soddisfano la relazione - ovvero $(\mathcal{M}_1, \dots, \mathcal{M}_n) \in \mathcal{R}_i^M \iff S \vdash \mathcal{R}_i(t_1, \dots, t_n)$
- l'interpretazione di un simbolo di funzione f è l'associazione $t_1, \dots, t_n \mapsto f(t_1, \dots, t_n)$ - $f_j^M : M^n \rightarrow M$ è t.c. $f_j^M(t_1, \dots, t_n) := f_j(t_1, \dots, t_n)$
es. $+^M(0, 1) = \underbrace{+}_{t}(0, 1)$

Osserviamo che l'interpretazione in \mathcal{M} di un termine chiuso t coincide con il termine stesso ($t^M = t$).

Vogliamo dimostrare $\mathcal{M} \models S$. Per farlo (come spesso accade) ci è più comodo dimostrare un'affermazione più forte: $\forall E \mathcal{M} \models E \iff S \vdash E$.

proof

Dimostriamo per casi (su E):

- $E \models \neg G$
 - se $\mathcal{M} \not\models E$, per ipotesi induttiva si ha $S \not\models G$ e da S completa segue $S \vdash \neg G$.
 - se $\mathcal{M} \models E$, allora per ipotesi induttiva $S \vdash G$ e da S coerente segue $S \not\models \neg G$
- $\mathcal{M} \models G \wedge H \iff S \vdash G \wedge H$
 - (\Rightarrow) supponiamo $\mathcal{M} \models G \wedge H$;
 - allora, $\mathcal{M} \models G \wedge \mathcal{M} \models H$
 - per ipotesi induttiva, abbiamo $S \vdash G \wedge S \vdash H$, e usando la tautologia $G \rightarrow (H \rightarrow (G \wedge H))$ ottengo $S \vdash G \wedge H$
 - (\Leftarrow) supponiamo $S \vdash G \wedge H$;
 - per la tautologia $G \wedge H \rightarrow G$, otteniamo $S \vdash G \wedge S \vdash H$
 - per ipotesi induttiva, segue $S \models G \wedge S \models H$, da cui $S \models G \wedge H$
- se E è un enunciato atomico $R(t_1, \dots, t_k)$, abbiamo che $\mathcal{M} \models R(t_1, \dots, t_n)$ per definizione $\iff (t_1^M, \dots, t_n^M) \in \mathbb{R}^M$ e, per l'osservazione sui termini, $\iff (t_1, \dots, t_n) \in R^M$, il che per definizione equivale a $S \vdash R(t_1, \dots, t_n)$
- se E è del tipo $\forall x F$
 - sappiamo $\iff \forall m \in M \mathcal{M} \models F[x/m]$
 - e qui vale $\mathcal{M} \models F[x/m] \iff \mathcal{M} \models F[x/t]$
 - per ipotesi induttiva, abbiamo che $\forall m \in M, S \vdash F[x/m]$, ma non abbiamo il \iff .
 - sappiamo $S \vdash \forall x F \implies S \vdash F[x/t]$, ma non l'altro verso (è come dire che sappiamo che per ogni numero c'è una dimostrazione, ma vogliamo che ci sia una dimostrazione per ogni numero)

Possiamo definire una teoria che ci aiuta nella dimostrazione.

Def. 35: Teoria con testimoni

Una **teoria con testimoni**, o **teoria di Henkin** (o “scapegoat theory”) soddisfa la seguente proprietà:

- \forall formula F con un’unica variabile libera x , \exists un termine chiuso t t.c.

$$T \vdash \exists x \neg F(x) \rightarrow \neg F(t)$$

(t è un testimone dell’enunciato $\exists x \neg F(x)$)

(un testimone è un termine specifico (t) che la teoria “nomina” per concretizzare un’affermazione esistenziale - normalmente, se diciamo $\exists x P(x)$, non sappiamo chi sia x per cui P vale; se siamo in una teoria con testimoni, invece, abbiamo $\exists x P(x) \rightarrow P(t)$, sappiamo esattamente chi sia t)

Vediamo come una teoria con testimoni ci aiuta a dimostrare il caso $\forall x F(x)$ con $F(x)$ aperta.

(Dimostriamo che, se $\mathcal{M} \models \forall x F$ allora $T \vdash \forall x F$)

Per assurdo, supponiamo $\mathcal{M} \models \forall x F$ e $T \not\vdash \forall x F$.

Per completezza di T vale $T \vdash \neg E$, ovvero $T \vdash \exists x \neg F(x)$. Dato che T è una teoria con testimoni, esiste un termine chiuso t tale che $T \vdash \exists x \neg F(x) \rightarrow \neg F(t)$.

Dunque $T \vdash \neg F(t)$.

Per ipotesi induttiva, $\mathcal{M} \models \neg F(t)$. Ma da $\mathcal{M} \models \forall x F(x)$, e da $\mathcal{M} \models \forall x F(x)$, vale, per qualsiasi termine chiuso $\mathcal{M} \models F(x) \rightarrow F(t)$.

Dunque, $\mathcal{M} \models F(t)$, il che contraddice $\mathcal{M} \models \neg F(t)$. □

Se T coerente, T si può estendere a $T^* \supseteq T$ che sia coerente e con testimoni (in un \mathcal{L} che estende \mathcal{L}_T con un insieme numerabile di costanti).

Thm. 26

Per ogni teoria T coerente esiste una teoria T' tale che:

- T' è un’estensione di T
- T' è una teoria con testimoni
- il linguaggio di T' è numerabile ed estende quello di T
- T' è coerente

proof

Sia T coerente.

Sia $T_0 = T \cup \{\text{istanze degli assiomi logici nel linguaggio esteso}\}$.

Fissiamo un’enumerazione delle formule in \mathcal{L} con una variabile libera: $F_1(x_1), F_2(x_2), \dots$

Sia $B = \{b_1, b_2, b_3, \dots\}$ un insieme di nuovi simboli di costante. Fissiamone un’enumerazione b_{j1}, b_{j2}, \dots , in cui b_{jk} è il primo (per indice) t.c.

- b_{jk} non appare in $F_1(x_1), \dots, F_k(x_k)$
- b_{jk} è diverso da b_{j1}, \dots, b_{jk-1}

quindi b_{j1} è la prima costante $\notin F_1$, b_{j2} la prima che non compare in F_2 e non è già stata usata (quindi $\neq b_{j1}$), ecc

Sia W_k il seguente enunciato:

$$\exists x_k \neg F_k(x_k) \rightarrow \neg F_k(b_{jk})$$

Sia $T_n = T_0 \cup \{W_1, \dots, W_n\}$, e sia $T_\infty = \bigcup_n T_n$.

claim: $T^* = \bigcup_{n \in \mathbb{N}} T$ è di Henkin, coerente ed estende T .

di Henkin: $\forall n \exists t = b_{jn}$ t.c. $T^* \ni \exists x_n \neg F_n(x_n) \rightarrow \neg F_n(b_{jn})$ è vero per costruzione

- Dimostriamo che T_∞ è coerente (basta dimostrare che $\forall n T_n$ coerente)

per induzione:

- **C.B.:** T_0 è coerente

“banale”, T coerente e abbiamo aggiunto a T solo assiomi

$T_0 \vdash \perp \implies T \cup A = \{A_1, \dots, A_t\} \vdash \perp$ (assiomi) $\implies T \vdash A \rightarrow \perp$ - visto che A assiomi, $T \vdash A$ e per MP, $T \vdash \perp$ - ma questo è impossibile perché T coerente)

- **P.I.:** assumiamo T_{n-1} coerente

per assurdo, sia T_n incoerente;

allora, si ha $T_n \vdash \perp$, e $\perp \rightarrow B \implies T_n \vdash B \ \forall B$; quindi $T_n \vdash \neg W_n$;

ma $T_n = T_{n-1} \cup \{W_n\}$, quindi (per deduzione) $T_{n-1} \vdash W_n \rightarrow \neg W_n$;

$\neg W_n$ è una negazione di un’implicazione (quindi verifica la premessa $\exists x_n \neg F_n(x_n)$ e falsifica la conseguenza $\neg F_n(b_{jn})$) quindi $T_{n-1} \vdash \exists x_n \neg F_n(x_n)$ e $T_{n-1} \vdash F_n(b_{jn})$.

Per costruzione, sappiamo che $T_{n-1} = T_0 (= T \cup \text{assiomi}) \cup \{W_1, \dots, W_{n-1}\}$; abbiamo scelto b_{jn} e quindi sappiamo che:

- esso non compare in W_1, \dots, W_{n-1} ;
- per definizione, non compare in T ;

Dovrà quindi far parte degli assiomi logici.

Notiamo che quindi, sostituendo b_{jn} con una nuova variabile, la formula restante rimarrà un assioma logico.

Sia quindi y una variabile che non compare nella dimostrazione δ (di $F_n(b_{jn})$).

claim: $\delta^y = \delta[b_{jn}/y] = (D_1^y, \dots, D_n^y)$ è ancora una dimostrazione.

Infatti:

- gli assiomi rimangono tali (o non dicono niente su b_{jn}) o dicono cose sempre vere, quindi sostituendo non cambia nulla
- le ipotesi T_{n-1}, W_1, \dots, W_n non cambiano, in quanto $\not\ni b_{jn}$
- il Modus Ponens è preservato
- la Generalizzazione non viene mai applicata su b_{jn} (vorrebbe dire che una formula $\forall y H(y)$ sarebbe stata ottenuta da una formula $H(b_{jn})$ (ricordiamo che b_{jn} è una costante specifica), il che è impossibile)

Ora - se è vero $T_{n-1} \vdash F_n(y)$, per Generalizzazione si ha anche $T_{n-1} \vdash \forall y F_n(y)$. Ma abbiamo anche $T_{n-1} \vdash \exists x \neg F_n(x)$. Le due cose sono chiaramente contraddittorie, quindi T_{n-1} è incoerente.

□

Quindi, possiamo formalizzare i seguenti teoremi:

Thm. 27

Sia T una teoria con testimoni e coerente in un linguaggio numerabile. Allora, T ha un modello numerabile.

Thm. 28: Esistenza del modello

Ogni teoria coerente (in un linguaggio numerabile) ha un modello numerabile.

proof

Partiamo da T coerente (in \mathcal{L})

- $\xrightarrow{\text{Henkin}} T^* \supseteq T$ coerente con testimoni (in \mathcal{L}^* numerabile $\supseteq \mathcal{L}$)
- $\xrightarrow{\text{Lindenbaum}} \hat{T} \supseteq T^*$ coerente e completa con testimoni (in \mathcal{L}^*)
(la proprietà di essere con testimoni è preservata perché il lemma di Lindenbaum non cambia il linguaggio)
- $\Rightarrow \mathcal{M}_{\hat{T}} \models T$ (il modello dei termini di \hat{T} è un modello numerabile di T)

nota bene ! logica con identità

La dimostrazione sopra descritta si limita alla logica di primo ordine senza identità.

Per il modello sopra definito, $\mathcal{M} \models (t_1 = t_2) \iff t$ ed s sono esattamente lo stesso termine (quindi, per esempio, $(1 + 1) \neq 2$).

Per risolvere questo problema, basta quoziare sulla seguente relazione:

$$t \sim s \iff T \vdash (t = s)$$

- notiamo che \sim è una relazione di equivalenza

Definiamo quindi il modello dei termini \mathcal{M}/\sim in questo modo:

- il dominio è $M/\sim = \{[t]_\sim \mid t$ termine chiuso $\}$
- $c^{\mathcal{M}/\sim} = [c]_\sim$
- $f^{\mathcal{M}/\sim}([t_1]_\sim, \dots, [t_n]_\sim) = [f(t_1, \dots, t_n)]_\sim$
- $([t_1]_\sim, \dots, [t_n]_\sim) \in R^{\mathcal{M}/\sim}$ se e solo se $T \vdash R(t_1, \dots, t_n)$

La dimostrazione $\mathcal{M}/\sim \models E \iff T \vdash E$ procede esattamente come quella di \mathcal{M} . Inoltre, se \mathcal{M} ha cardinalità numerabile, anche \mathcal{M}/\sim ha cardinalità numerabile.

4.1. Teorema di compattezza

Il teorema di compattezza si può riformulare in diversi modi:

Thm. 29: Teorema di compattezza, versioni equivalenti

- Un insieme di enunciati è coerente se e solo se ogni suo sottinsieme finito è coerente.
- Un insieme di enunciati ha un modello se e soltanto se ogni suo sottoinsieme finito ha un modello.

- $T \models E$ se e solo se esiste un sottoinsieme finito $T_0 \subseteq T$ tale che $T_0 \models E$.

4.1.1. Applicazione: (non) assiomatizzabilità

Data una proprietà P di strutture, è utile chiedersi se possa essere espressa da enunciati del primo ordine, ovvero se esista un insieme di enunciati T che soddisfa, per ogni struttura \mathcal{A} , la seguente equivalenza:

$$\mathcal{A} \models T \iff \mathcal{A} \text{ ha la proprietà } P$$

Se un tale T esiste, diciamo che **assiomatizza** o *definisce* la proprietà P .

Ha ancora più senso chiedersi, nello specifico, data una classe di strutture \mathcal{C} e un linguaggio predicativo \mathcal{L} , se esista una teoria in \mathcal{L} tale che la proprietà P sia soddisfatta.

Se esiste una teoria T che assiomatizza una proprietà P , ci si può chiedere se esiste un *insieme finito di enunciati* che assiomatizza P . Questo è equivalente a chiedersi se esiste un singolo enunciato E (l'AND tra tutti gli enunciati) tale che, per ogni struttura \mathcal{A} nella classe \mathcal{C} ,

$$A \models E \iff \mathcal{A} \text{ ha la proprietà } P$$

In questo caso, diciamo che P è **finitamente assiomatizzabile** relativamente alla classe \mathcal{C} .

Esempio 1: finitezza

- $P = \text{"avere dominio finito";}$
- $\mathcal{C} = \text{tutte le strutture;}$

Partiamo da $P_n = \text{"avere dominio di cardinalità } \geq n\text{"}$.

Possiamo assiomatizzare questa proposizione con un linguaggio \mathcal{L} “vuoto” (senza relazioni specifiche) in questo modo:

$$\exists x_1, \dots, x_n \left(\bigwedge_{i \neq j, 1 \leq i \leq n} \neg(x_i = x_j) \right)$$

Possiamo assiomatizzare anche la proprietà “avere dominio di cardinalità n ” in questo modo:

$$\exists x_1, \dots, x_n \left(\bigwedge_{i \neq j, 1 \leq i \leq n} \neg(x_i = x_j) \wedge \forall y \left(\bigvee_{i=1}^n (y = x_i) \right) \right)$$

Prop. 2: Assiomatizzazione di “avere un dominio finito / infinito”

La proprietà “avere dominio infinito” è assiomatizzabile ma non finitamente assiomatizzabile.

La proprietà “avere dominio finito” non è assiomatizzabile.

Consideriamo $T = \{P_n \mid n \in \mathbb{N}\}$ (“avere dominio di cardinalità almeno n ” per ogni n). La teoria T assiomatizza “avere un dominio infinito” (se si ha dominio almeno n per ogni n)

Sia per assurdo F una teoria che assiomatizza $P = \text{"avere dominio finito”}$.

Consideriamo $F \cup T$. $F \cup T$ non può avere modelli (in quanto parla di essere finito e infinito allo stesso tempo).

Se ne prendo un qualsiasi pezzo finito $T_0 \subseteq F \cup T$, noto però che esso ha un modello (in quanto descrive, “alla peggio”, l’avere dominio finito (da F) e l’avere almeno n elementi (da T)).

Visto che T_0 ha un modello, per compattezza tutta la teoria dovrebbe avere un modello ($\text{FINSAT} \implies \text{SAT}$), il che è impossibile. \square

Vediamo anche che è impossibile che “avere dominio infinito” sia *finitamente* assiomatizzabile.

Infatti, se esistesse S finita t.c. $\mathcal{A} \models S \iff \mathcal{A}$ ha dominio infinito, avremmo che $\neg S$ assiomatizzerebbe “avere dominio finito” (impossibile). \square

Lemma 7

Sia P assiomatizzabile. Se $\neg P$ non è assiomatizzabile, allora P non è finitamente assiomatizzabile.

Per contrapposizione, se $P, \neg P$ sono assiomatizzabili, allora P è finitamente assiomatizzabile.

Lemma 8

Se P è assiomatizzabile da T e P è finitamente assiomatizzabile, allora $\exists T_0 \subseteq T$ che assiomatizza P .

Lemma 9

Se T ha modelli finiti arbitrariamente grandi (per ogni n c’è un modello con almeno n elementi), allora T ha anche modelli infiniti.

(Non) assiomatizzazione di “essere un grafo connesso”

Consideriamo i grafi, con linguaggio $\mathcal{L} = \{E(x, y)\}$, e la proprietà P = “essere un grafo connesso”.

Possiamo scrivere la proprietà D_n = “essere a distanza $\geq n$ ” in questo modo:

$$\neg(\exists x_1, \dots, x_n \left(\bigwedge_{i \neq j} \neg(x_i = x_j) \wedge E(c, x_1) \wedge E(x_1, x_2) \wedge \dots \wedge E(x_n, d) \right))$$

Supponiamo che P sia assiomatizzabile da una teoria C .

Consideriamo $C \cup \{D_n \mid n \geq 0\}$ - questa teoria non può avere un modello.

Come prima, consideriamo un pezzo finito $T_0 \subseteq C \cup \{D_n \mid n \geq 0\}$.

Come prima, questo sottoinsieme ha un modello (un grafo connesso con vertici a distanza $>$ del massimo n). Come prima, avremmo $\text{FINSAT} \implies \text{SAT}$, e per questo si ha che P non è assiomatizzabile. \square

4.1.2. Modelli non standard dell’aritmetica

Dato $\mathcal{N} = (\mathbb{N}, 0, 1, +, *, \leq)$, possiamo definire la sua teoria $T = \text{Th}(\mathcal{N}) = \{E \mid \mathbb{N} \models E\}$.

Sia $\mathcal{A} \models \text{Th}(\mathcal{N})$. Quanto assomiglia \mathcal{A} ad \mathbb{N} ?

Noi vorremmo $\mathcal{A} \cong \mathcal{N}$, ma in realtà esistono molti modelli di \mathcal{N} molto differenti dai numeri naturali che conosciamo.

Consideriamo per esempio una nuova costante c e la teoria:

$$T \cup \{A_n \mid n \in \mathbb{N}\}$$

con $A_n = \underbrace{1 + \dots + 1}_n < c$.

Un sottoonsieme qualsiasi $T_0 \subseteq T \cup \{A_n \mid n \in \mathbb{N}\}$ ha un modello - basta dare un'interpretazione alla costante c .

Per esempio, $\mathcal{A} = (\mathbb{N}, +^{\mathcal{N}}, x^{\mathcal{N}}, 0^{\mathcal{N}}, 1^{\mathcal{N}}, c^{\mathcal{A}} = \max(a_1, \dots, a_k) + 1)$ è un modello di T_0 .

Per compattezza, quindi, $T \cup \{A_n \mid n \in \mathbb{N}\}$ ha un modello.

Ma come interpretiamo $c^{\mathcal{A}}$ in questo modello? Deve esserci un unico c tale che, $\forall n, \underbrace{1 + \dots + 1}_n <^{\mathcal{A}} c^{\mathcal{A}}$.

Deve quindi esserci un c maggior di tutti gli elementi di \mathbb{N} .

Questo modello è molto diverso da \mathbb{N} . \mathcal{A} è quello che si dice **modello non-standard** dell'aritmetica, e $c^{\mathcal{A}}$ è un *naturale non-standard*.

Inizia infatti con \mathbb{N} , ma contiene almeno un elemento maggiore di tutti i numeri standard. Inoltre, se $a \in \mathcal{A}$ è un numero non-standard, allora anche il suo predecessore è un numero non-standard. Un elemento non-standard ha quindi infiniti predecessori e successori, nessuno dei quali può essere standard.

Intorno a $c^{\mathcal{A}}$ si sviluppa quindi una copia isomorfa a \mathbb{Z} .

Si esclude quindi che si possa definire una teoria T che assiomatizzi $Th(\mathcal{N})$ in modo che valga, come per DLO $<$ che tutti i modelli numerabili di T sono isomorfi. Resta aperta però la possibilità di trovare un insieme di assiomi T con insieme di teoremi computabilmente enumerabili che coincida con la "vera" Teoria dei Numeri $Th(\mathcal{N})$. Una tale teoria sarebbe completa e dunque decidibile e fornirebbe un algoritmo per decidere automaticamente se un enunciato E è un teorema della Teoria dei Numeri o no.

Vademecum su limiti espressivi e cardinalità

<i>proprietà del Modello M</i>	<i>assiom.?</i>	<i>causa</i>
Essere finito	NO	Compattezza. Se T ha modelli finiti arbitrariamente grandi, ne ha anche uno infinito.
Essere numerabile (esattamente \aleph_0)	NO	LS Up. Se ha un modello infinito numerabile, ne ha anche uno più che numerabile.
Essere più che numerabile ($> \aleph_0$)	NO	LS Down. Se ha un modello più che numerabile (in ling. numerabile), ne ha anche uno numerabile.
Essere ben ordinato	NO	Compattezza. I modelli non-standard contengono sempre copie di \mathbb{Z} o parti dense.

Tabella 4.1: Proprietà NON Assiomatizzabili (Limiti Espressivi)

<i>proprietà del Modello M</i>	<i>assiom.?</i>	<i>come ?</i>
Avere almeno n elementi	SÌ	$\exists x_1 \dots x_n (\bigwedge_{i \neq j} x_i \neq x_j)$
Avere al più n elementi	SÌ	$\exists x_1 \dots x_n \forall y (\bigvee_{i=1}^n y = x_i)$
Avere esattamente n elementi	SÌ	Congiunzione di "almeno n " e "al più n ".
Essere infinito	SÌ	Con una teoria infinita $T_{inf} = \{"\text{almeno } n" \mid n \in \mathbb{N}\}$.

Tabella 4.2: Proprietà Assiomatizzabili in Logica del Primo Ordine

I teoremi di Gödel

5.1. Funzioni calcolabili (algoritmiche)

Esistono diverse definizioni di algoritmo. Le principali ci vengono date da Gödel, Herbrand e Kleene (in chiave matematica), da Turing (attraverso le TM), e da Church (nel λ -calcolo).

Def. 36: Funzioni calcolabili

La classe \mathcal{C} delle funzioni parziali calcolabili è la minima classe di funzioni del tipo $\mathbb{N}^k \rightarrow \mathbb{N}$ (con $k \in N$) t.c.:

- $+, * \in \mathcal{C}$
- $i(x, y) = \begin{cases} 1 & \text{se } x = y \\ 0 & \text{se } x \neq y \end{cases}$
- $\prod_i^n(x_1, \dots, x_n) = x_i$ (proiezione)

e \mathcal{C} chiusa sotto composizione - date $\theta_1, \dots, \theta_n : \mathbb{N}^k \rightarrow \mathbb{N}, \psi : \mathbb{N}^m \rightarrow \mathbb{N}$, la composta ϕ è definita come segue:

$$\phi(x_1, \dots, x_n) = \psi(\theta_1(x_1, \dots, x_n), \dots, \theta_n(x_1, \dots, x_k))$$

\mathcal{C} è chiusa anche sotto *minimalizzazione*: data $g(\vec{x}, y)$, la funzione $h(\vec{x})$ è definita come $h(\vec{x}) = \min z$ t.c.

- i valori $g(\vec{x}, 0), g(\vec{x}, 1), \dots, g(\vec{x}, z - 1)$ sono definiti e $\neq 0$
- $g(\vec{x}, z) = 0$
- (se un tale z non esiste, la funzione non è definita)

Prop. 3

Si ha che $f \in \mathcal{C} \implies f$ calcolabile.

- Nella composizione di funzioni $h(g_1, \dots, g_m)$:
 - Se tutte le g_i e h fossero totali, si applicherebbero semplicemente le g_i e poi h sul risultato.
 - Se una delle g_i è indefinita, il calcolo procede comunque secondo la logica algoritmica: il programma semplicemente non termina (non si arriva mai ad applicare h).
- (anche se il risultato è indefinito, il processo rimane algoritmico).

Per calcolare una composizione come $h(g_1(\vec{x}), g_2(\vec{x}), \dots)$, è necessario stabilire un ordine di esecuzione:

- Non si possono far partire le funzioni in parallelo; devono essere eseguite in ordine sequenziale.
 - se si procedesse in parallelo, si potrebbe trovare un risultato per g_2 mentre g_1 sta ancora calcolando (o è in loop infinito).
- Per definizione, se un argomento è indefinito, l'intera composizione deve esserlo. Procedendo in ordine, se ci si "ferma" su g_1 , il calcolo si arresta correttamente senza produrre risultati parziali o errati basati sugli altri argomenti.

(Anche $g(\vec{x}, y)$ è algoritmica perché lo è $h(\vec{x})$)

Prop. 4: Calcolabilità

- Una funzione $\varphi : \mathbb{N}^k \rightarrow \mathbb{N}$ è calcolabile $\iff \varphi \in \mathcal{C}$.
- $R \subseteq \mathbb{N}^k$ si dice calcolabile se la sua **funzione caratteristica** χ_R è calcolabile.

$$\chi_R(\vec{x}) = \begin{cases} 1 & \text{se } \vec{x} \in R \\ 0 & \text{se } \vec{x} \notin R \end{cases}$$

5.2. Teorema di Definibilità

Def. 37: Definibilità

Diciamo che φ è **rappresentabile** in \mathcal{N} da una formula $F(x_1, \dots, x_k, y)$ se per ogni tupla di numeri naturali $(a_1, \dots, a_k, b) \subseteq \mathbb{N}^{k+1}$, vale l'equivalenza:

$$\varphi(a_1, \dots, a_k) = b \iff \mathcal{N} \models F\left[\left(\begin{smallmatrix} x_1, \dots, x_k, y \\ a_1, \dots, a_k, b \end{smallmatrix}\right)\right]$$

i.e. se le $k+1$ -ple (a_1, \dots, a_k, b) appartententi al grafo di φ sono esattamente quelle che soddisfano in \mathcal{N} la formula $F(x_1, \dots, x_k, y)$ assegnando a_i a x_i e b a y .

(la formula è vera nel modello standard \mathcal{N} se e solo se la funzione, calcolata sugli input a_i , restituisce b).

Nel linguaggio dell'aritmetica, ogni numero naturale $n \in \mathbb{N}$ ha un nome canonico (**numerale**) costituito dalla somma di n volte la costante 1.

Quindi, vale che:

$$\mathcal{N} \models F\left[\left(\begin{smallmatrix} x_1, \dots, x_k, y \\ a_1, \dots, a_k, b \end{smallmatrix}\right)\right] \iff \mathcal{N} \models F(\bar{a}_1, \dots, \bar{a}_k, \bar{b})$$

Thm. 30: Teorema di Definibilità

Le funzioni calcolabili sono definibili in \mathcal{N} .

per induzione

- **C.B.:** le funzioni di base sono definibili in \mathcal{N}

L'addizione è definita dalla formula $F(x, y, z) := ((x + y) = z)$, la moltiplicazione dalla formula $G(x, y, z) := ((x \times y) = z)$, la proiezione $\pi_i^n(x_1, \dots, x_n) = x_i$ (dove $i \in [1, n]$) è definibile dalla formula $H(x_1, \dots, x_n, z) := (x_1 = x_1 \wedge \dots \wedge x_i = z \wedge \dots \wedge x_n = x_n)$, la funzione caratteristica dell'uguaglianza dalla formula $I(x, y, z) := (x = y \wedge z = 1) \vee (x \neq y \wedge z = 0)$. \square

■ **chiusura per composizione:** le funzioni definibili in \mathcal{N} sono chiuse per composizione

Siano:

$$\begin{array}{ll} H_1(x_1, \dots, x_k, y_1) & \text{che rappresenta } \vartheta_1 \\ H_2(x_1, \dots, x_k, y_2) & \text{che rappresenta } \vartheta_2 \\ \vdots & \\ H_m(x_1, \dots, x_k, y_m) & \text{che rappresenta } \vartheta_m \end{array}$$

E sia:

$$G(y_1, \dots, y_m, z) \quad \text{che rappresenta } \Psi$$

Voglio una formula $F(x_1, \dots, x_k, w)$ che rappresenti la funzione composta $\Psi(\vartheta_1, \dots, \vartheta_m)$.

Quand'è che $(x_1, \dots, x_k, w) \in$ grafico composta($\Psi, \vartheta_1, \dots, \vartheta_m$)? Se W è il valore di Ψ con argomenti = m valori di $\vartheta_1, \dots, \vartheta_m$ applicate a (x_1, \dots, x_k) .

La formula è:

$$\exists y_1 \dots y_m \quad (\text{se esistono gli } m \text{ valori intermedi})$$

tale che:

$$\begin{aligned} & H_1(x_1 \dots x_k, y_1) \wedge H_2(x_1 \dots x_k, y_2) \wedge \dots \wedge H_m(x_1 \dots x_k, y_m) \\ & \wedge \quad G(y_1, \dots, y_m, W) \quad (\text{valore } G(y_1 \dots y_m)) \end{aligned}$$

NB: Questo finisce nel grafico della composta se $\exists y_1 \dots$ (se le intermedie sono indefinite, anche la composta lo è).

Si vede che questa formula è soddisfatta \iff l'elemento appartiene al grafico della composta.
siano

■ **chiusura per minimo:** le funzioni definibili in \mathcal{N} sono chiuse per minimo

Sia $H(x_1, \dots, x_k, y, z)$ una formula che rappresenta $\vartheta(x_1, \dots, x_k, y)$.

Voglio G con variabili (x_1, \dots, x_k, S) dove S è il minimo tale che:

$$\vartheta(x_1, \dots, x_k, S) \downarrow = 0 \quad \wedge \quad \forall w (0 \leq w < S, \vartheta(x_1, \dots, x_k, w) \downarrow \neq 0)$$

La formula risultante è:

$$H(x_1, \dots, x_k, S, 0) \wedge \forall w (w < S \rightarrow \exists z (H(x_1, \dots, x_k, w, z) \wedge z \neq 0))$$

Def. 38: Calcolabilità e rappresentabilità di una relazione

Una relazione $R \subseteq N^n$ è calcolabile se e solo se la sua funzione caratteristica è calcolabile.

Una relazione R è rappresentabile in \mathcal{N} se e solo se $\exists F(x_1, \dots, x_k)$ t.c. $\forall n_1, \dots, n_k \in \mathbb{N}$:

$$\begin{aligned} (n_1, \dots, n_k) \in R &\implies \mathcal{N} \models F(\bar{n}_1, \dots, \bar{n}_k) \\ (n_1, \dots, n_k) \notin R &\implies \mathcal{N} \models \neg F(\bar{n}_1, \dots, \bar{n}_k) \end{aligned}$$

Thm. 31: Rappresentabilità di relazioni

Tutte le relazioni calcolabili sono rappresentabili in \mathcal{N}

R è calcolabile se e solo se $\chi_R \in \mathcal{C}$.

Sia $F(x_1, \dots, x_k, y)$ che rappresenta χ_R .

Questo significa che $\chi_R(n_1, \dots, n_k) = b \iff \mathcal{N} \models F(\bar{n}_1, \dots, \bar{n}_k, \bar{b})$. Abbiamo che $b \in \{0, 1\}$.

I vettori che appartengono a R sono quelli che danno 1 (per come definiamo χ_R), quindi $F(x_1, \dots, x_k, 1)$ rappresenta R .

Ovvero, presi $(n_1, \dots, n_k) \in R$, $\chi_R(n_1, \dots, n_k) = 1 \implies \mathcal{N} \models F(\bar{n}_1, \dots, \bar{n}_k, 1)$

Se invece $(n_1, \dots, n_k) \notin R$, $\chi_R(n_1, \dots, n_k) = 0 \implies \mathcal{N} \models F(\bar{n}_1, \dots, \bar{n}_k, 0) \implies \mathcal{N} \models \neg F(\bar{n}_1, \dots, \bar{n}_k, 1)$

L'implicazione finale vale perché:

- $\mathcal{N} \models \neg(0 = 1)$
- $\mathcal{N} \models "F \text{ è funzionale}"$

(dato che F rappresenta una funzione, non ci possono essere 2 valori diversi per cui sia vera sullo stesso parametro)

Osservazione: notiamo che tutte le funzioni calcolabili hanno una simile struttura sintattica: sono formate da quantificatori esistenziali seguiti da una formula in cui non compaiono quantificatori o essi appaiono solo “limitati” (es. $\forall x(x \leq 7 \rightarrow \dots)$) - queste formule vengono dette \sum_1^0 .

5.3. I numeri di Gödel

Vogliamo mostrare che $Th(\mathcal{N})$ non è calcolabile mostrando che non è rappresentabile in \mathcal{N} .

Funzioni e relazioni calcolabili hanno come argomenti numeri naturali. Ci serve quindi poter codificare funzioni e relazioni su altri tipi di oggetti.

Gödel inventa una codifica $code : \{\text{enunciati aritmetici}\} \rightarrow \mathbb{N}$ tale che:

- ad ogni simbolo di base del linguaggio (parentesi, connettivi, variabili, simboli di funzione/relazione, costanti) viene associato un **intero positivo dispari**
- ogni formula ben formata è una sequenza u_0, u_1, \dots, u_n di simboli di base, che viene codificata in questo modo:

$$code(u_0, u_1, \dots, u_n) = 2^{code(u_0)} \cdot 3^{code(u_1)} \cdot \dots \cdot p_n^{code(u_n)}$$

- ogni derivazione è una sequenza e_0, e_1, \dots, e_n di formule, che viene codificata in questo modo:

$$code(e_0, e_1, \dots, e_n) = 2^{code(e_0)} \cdot 3^{code(e_1)} \cdot \dots \cdot p_n^{code(e_n)}$$

La funzione $code$ è iniettiva e permette di distinguere algoritmamente tra codici di simboli di base (pari), “formule” (pari con primo esponente dispari) e “derivazioni” (pari con primo esponente pari).

Quando si parla dell'esprimibilità di un insieme di enunciati in una teoria, si intende l'esprimibilità dell'insieme dei codici numerici enunciati.

Ci chiediamo quindi se $\{cod(E) \mid \mathcal{N} \models E\}$ sia rappresentabile in \mathcal{N} (non lo è).

5.4. Indecidibilità algoritmica dell'aritmetica (Tarski)

Thm. 32: **Teorema di indefinibilità di Tarski**

La verità aritmetica non può essere definita all'interno dell'aritmetica.

(L'insieme delle verità aritmetiche $Th(\mathcal{N}) = \{E \mid \mathcal{N} \models E\}$ non è rappresentabile in \mathcal{N})

Supponiamo per assurdo che $Th(\mathcal{N})$ sia rappresentabile. Si tratta di un insieme, quindi è descritto da una formula con una variabile libera.

Sia $T(x)$ la formula che lo rappresenta.

- $E \in Th(\mathcal{N}) \implies \mathcal{N} \models V(\overline{\text{code}(E)})$
- $E \notin Th(\mathcal{N}) \implies \mathcal{N} \models \neg V(\overline{\text{code}(E)})$

Introduciamo una nuova funzione $\delta : \mathbb{N} \rightarrow \mathbb{N}$. Dato $p \in \mathbb{N}$, δ controlla se p è il codice numerico di una formula $A(x)$ una variabile libera. Se lo è, $\delta(p) = \text{codice di } A(x/\bar{p})$.

(dato p , controllo se codifica una FML $A(x)$ - se sì, scrivo $A(p)$ (metto il numerale p al posto della variabile (quindi do alla formula il suo stesso codice)), codifico la nuova stringa e restituisco il suo codice numerico q)

δ è evidentemente calcolabile (uso la tesi di Church-Turing - ogni funzione descrivibile tramite un algoritmo è calcolabile da una TM - per non doverlo dimostrare esplicitamente). In quanto calcolabile, $\delta \in \mathcal{C}$, e quindi δ rappresentabile.

Quindi, sia $D(x, y)$ una formula che rappresenta δ in \mathcal{N} , ovvero tale che:

$$\delta(a) = b \iff \mathcal{N} \models D(\bar{a}, \bar{b})$$

Consideriamo la formula

$$\forall y(D(x, y) \rightarrow \neg V(y))$$

(che dichiara “per ogni y valore di $\delta(x)$ (quindi formula $A(y/\bar{x})$ in cui $x = \text{code}(A)$), y non è un enunciato vero in \mathcal{N} ”)

Questa formula è una formula con una variabile libera x (chiamiamola $A(x)$). (In quanto tale può essere codificata).

Sia p il codice di $A(x)$.

Consideriamo $A(x/\bar{p})$, ovvero

$$\forall y(D(\bar{p}, y) \rightarrow \neg V(y))$$

(che dichiara che ogni y valore di $\delta(p)$ non è un enunciato vero in \mathcal{N})

Sia q il suo codice (quindi $\delta(p) = q$).

Notiamo subito che la formula genera una contraddizione: la formula dichiara che ogni numero che sia $= \delta(p)$ non è vero in \mathcal{N} , ma essa stessa è uno di quei numeri (quindi sta dichiarando di “mentire” (di non essere vera)).

Dimostriamolo formalmente, ovvero che l'ipotesi che la verità esprimibile porti alla contraddizione $\mathcal{N} \models V(q) \wedge \mathcal{N} \models \neg V(q)$.

Dato che la formula D definisce la funzione δ in \mathbb{N} , abbiamo che

$$\mathbb{N} \models D(p, q)$$

Dimostriamo $\mathbb{N} \models \neg V(q)$:

■ **caso 1:** $\mathbb{N} \not\models A(p)$

$A(p)$ non è una verità in \mathbb{N} , dunque $code(A(p)) = q \notin Th(\mathbb{N})$, dunque per ipotesi $\mathbb{N} \models \neg V(q)$

■ **caso 2:** $\mathbb{N} \models A(p)$

se $A(p)$ è vero, è vero per $y = q$ ($\mathbb{N} \models D(p, q) \rightarrow \neg V(q)$), quindi, dato che $\mathbb{N} \models D(p, q)$ otteniamo $\mathbb{N} \models \neg V(q)$

Dimostriamo ora $\mathbb{N} \models V(q)$:

Da $\mathbb{N} \models D(p, q)$ e dalla funzionalità della formula D segue $\mathbb{N} \models (p, y) \rightarrow y = q$.

Visto che abbiamo appena dimostrato $\mathbb{N} \models \neg V(q)$, abbiamo (se non vale per q , sicuramente se $y = q$ non vale per y)

$$\mathbb{N} \models y = q \rightarrow \neg V(y)$$

Dunque, (poiché $\mathbb{N} \models (p, y) \rightarrow y = q$), abbiamo anche

$$\mathbb{N} \models D(p, y) \rightarrow \neg V(y)$$

Se vale con variabile libera allora vale anche nella forma

$$\mathbb{N} \models \forall y(D(p, y) \rightarrow \neg V(y))$$

che è esattamente $A(p)$.

Dunque, $A(p) \in Th(\mathbb{N})$, ovvero $A(p)$ è una verità aritmetica.

Quindi, dato che V rappresenta $Th(\mathbb{N})$, abbiamo

$$\mathbb{N} \models V(code(A(p))) = V(q)$$

il che è impossibile.

Abbiamo quindi:

Corollary for Thm. 32:

L'insieme delle verità aritmetiche $Th(\mathbb{N})$ non è algoritmicamente decidibile.

5.4.1. Prima forma debole di Incompletezza

Otteniamo quindi una prima forma debole del Primo Teorema di Incompletezza di Gödel

Thm. 33: Primo Teorema di Incompletezza (forma debole)

Se T è una teoria decidibile e tale che $\mathbb{N} \models T$, allora T non è completa: esiste un enunciato G tale che $T \not\models G \wedge T \not\models \neg G$

Se T fosse completa, dato che $\mathbb{N} \models T$ seguirebbe che $\{E \mid T \vdash E\} \subseteq \{E \mid \mathbb{N} \models E\}$. In più, per completezza, poiché abbiamo $T \vdash E \oplus T \vdash \neg E$, e poiché, visto che $\mathbb{N} \models T$, E è o vero o falso in \mathbb{N} , abbiamo

$$\{E \mid T \vdash E\} = \{E \mid \mathbb{N} \models E\} = Th(\mathbb{N})$$

Sapendo che $Th(\mathbb{N})$ non è decidibile, otteniamo che:

- se T decidibile e completa, allora i teoremi di T sono decidibili

- se T completa e $\mathbb{N} \models T$ allora i teoremi di T sono indecidibili
ovvero, se $\mathbb{N} \models T$, è impossibile che T sia sia completa che decidibile.

5.5. Primo Teorema di Incompletezza di Gödel

Thm. 34: Primo Teorema di Incompletezza

Se T è decidibile, coerente e contiene un minimo di aritmetica (MA, aritmetica minimale), allora T è incompleta.

NB: in realtà, Gödel ha dimostrato una versione più debole che assume la ω -coerenza (e non la coerenza)

La differenza principale di questa versione rispetto a quella precedente è che non si assume $\mathbb{N} \models T$: le ipotesi su T sono di natura sintattica e non fanno riferimento a strutture infinite o al concetto semantico di soddisfacibilità.

5.5.1. Aritmetica minimale (MA)

L'aritmetica minimale è assiomatizzata dai seguenti assiomi nel linguaggio $\{0, 1, +, \times\}$. Per ogni $n \in \mathbb{N}$ denotiamo con n il termine ottenuto sommando la costante 1 a se stessa n volte.

- (Ax 1) $0 + 1 = 1$
- (Ax 2) $\forall x(x + 1 \neq 0)$
- (Ax 3) $\forall x(x \neq 0 \rightarrow \exists z(z + 1 = x))$
- (Ax 4) $\forall x \forall y(x + 1 = y + 1 \rightarrow x = y)$
- (Ax 5) $\forall x(x + 0 = x)$
- (Ax 6) $\forall x \forall y(x + (y + 1) = (x + y) + 1)$
- (Ax 7) $\forall x(x \times 0 = 0)$
- (Ax 8) $\forall x \forall y(x \times (y + 1) = (x \times y) + x)$
- (Ax 9) $\forall x \forall y(x < y \vee x = y \vee y < x)$

Dobbiamo verificare che MA sia sufficiente a rappresentare tutte le funzioni calcolabili.

Def. 39: Rappresentabilità di una funzione in una teoria

Una funzione $\varphi : \mathbb{N}^k \rightarrow \mathbb{N}$ è rappresentabile in una teoria T se esiste una formula $F_\varphi(x_1, \dots, x_k, z)$ nel linguaggio di T tale che

$$T \vdash (\forall x_1) \dots (\forall x_k) (\forall y) (\forall z) [F_\varphi(x_1, \dots, x_k, y) \wedge F_\varphi(x_1, \dots, x_k, z) \rightarrow y = z],$$

e per ogni n_1, \dots, n_k, m in \mathbb{N}

$$\text{Se } \varphi(n_1, \dots, n_k) = m \text{ allora } T \vdash F_\varphi(\bar{n}_1, \dots, \bar{n}_k, \bar{m}).$$

(se l'ultima implicazione si inverte, si parla di *rappresentabilità forte*).

Thm. 35: Rappresentabilità in MA

Tutte le funzioni calcolabili sono rappresentabili in MA.

proof

La dimostrazione segue quella del Teorema di Rappresentabilità delle funzioni computabili in \mathbb{N} sostituendo gli argomenti semantici ($\mathbb{N} \models$) con argomenti sintattici MA (\vdash).

Def. 40: Rappresentabilità di una relazione in una teoria

Una relazione $R \subseteq \mathbb{N}^k$ è rappresentabile in una teoria T se esiste una formula $F_R(x_1, \dots, x_k)$ nel linguaggio di T con k variabili libere tale che

- (1) Se $(n_1, \dots, n_k) \in R$ allora $T \vdash F_R(\bar{n}_1, \dots, \bar{n}_k)$,
- (2) Se $(n_1, \dots, n_k) \notin R$ allora $T \vdash \neg F_R(\bar{n}_1, \dots, \bar{n}_k)$

Se ogni funzione è rappresentabile, ogni relazione è rappresentabile.

5.5.2. Il teorema

Gödel ha individuato una relazione binaria $R \subseteq \mathbb{N} \times \mathbb{N}$ tale che:

$$(m, n) \in R \iff m \text{ è il codice numerico di una formula } A \text{ con una variabile libera e} \\ n \text{ è il codice numerico di una dimostrazione in } T \text{ di } A(\bar{m})$$

Notiamo che questa relazione è **calcolabile** (poiché T è decidibile, posso controllare algoritmamente se una sequenza di formule è una dimostrazione valida). È quindi anche **rappresentabile** in T da una formula $F(x, y)$ tale che:

- $(m, n) \in R \implies T \vdash F(m, n)$
- $(m, n) \notin R \implies T \vdash \neg F(m, n)$

Consideriamo l'enunciato $A(x) \equiv \forall y \neg F(x, y)$ (che dice “non esiste una dimostrazione di me stesso”)

Sia m il codice di $A(x)$.

Abbiamo che $G = A(\bar{m}) \equiv \forall y \neg F(\bar{m}, y)$, ovvero “per ogni y , y non è il codice di una dimostrazione della formula che ha codice m ” (ovvero $A(\bar{m})$ stessa).

Lemma 10

G non è dimostrabile né refutabile in T , ovvero:

$$T \not\vdash \forall y \neg F(\bar{m}, y)$$

$$T \not\vdash \neg(\forall y \neg F(\bar{m}, y))$$

- (1) $T \not\vdash \forall y \neg F(\bar{m}, y)$

Supponiamo per assurdo T coerente e $T \vdash \forall y \neg F(\bar{m}, y)$.

Vuol dire che ne esiste una dimostrazione con derivazione formale. Sia n il codice di una tale

derivazione. Poiché n è la dimostrazione di G , si ha $(m, n) \in R$.

Poiché R è rappresentata da F , otteniamo che $T \vdash F(\bar{m}, n)$, quindi

$$T \vdash \exists y F(\bar{m}, y)$$

che è esattamente l'opposto dell'ipotesi $T \vdash \forall y \neg F(\bar{m}, y)$ (ovvero avremmo T incoerente)

(2) $T \not\vdash \forall y \neg F(\bar{m}, y)$

Supponiamo per assurdo $T \vdash \neg(\forall y \neg F(\bar{m}, y)) \equiv \exists y F(\bar{m}, y)$. (ovvero, in termini intuitivi, “Esiste un numero y che è il codice della dimostrazione di G ”)

Dalla prima parte del teorema (basata sulla semplice coerenza), sappiamo però che G non è dimostrabile. Pertanto, nessun numero naturale $n \in \mathbb{N}$ corrisponde al codice di una sua dimostrazione. Poiché la teoria T è corretta nel verificare le proprietà calcolabili (rappresentabilità forte), per ogni specifico numero n la teoria dimostra che esso non è la dimostrazione cercata:

$$\forall n \in \mathbb{N}, \quad T \vdash \neg F(m, n)$$

Sorge questa situazione apparentemente paradossale:

- La teoria dice $\exists y$
- Ma dice anche $\forall n \in \mathbb{N}, \quad T \vdash \neg F(m, n)$

La semplice coerenza logica non vieta questa situazione (la teoria potrebbe riferirsi a un elemento “infinito” non standard). Per escludere questo caso, Gödel introduce la ω -coerenza.

Def. 41: ω -coerenza

Una teoria T è ω -coerente se non accade mai che:

$$T \vdash \exists x \varphi(x) \quad \text{e contemporaneamente} \quad \forall n \in \mathbb{N}, \quad T \vdash \neg \varphi(n)$$

Se assumiamo che T sia ω -coerente, la situazione descritta sopra è impossibile. Poiché $\forall n \in \mathbb{N}, \quad T \vdash \neg F(m, n)$ è un fatto vero e inevitabile, dobbiamo rifiutare l'ipotesi iniziale $T \vdash \exists y F(m, y)$. Quindi, se T è ω -coerente, $\neg G$ non è dimostrabile in T .

Possiamo quindi formulare più accuratamente il Primo Teorema di Incompletezza:

Thm. 36: Primo Teorema di Incompletezza

Sia T una teoria formale che estende MA.

- se T è coerente, allora G è indimostrabile in T
- se T è ω -coerente, allora $\neg G$ è indimostrabile in T

Dunque, se T è ω -coerente, T è incompleta.