

Logica Matematica

Sapienza Università di Roma

libro del corso: tbd

aglaia norza

8 dicembre 2025

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 | 6 |
| 1.5 Forme normali | 7 |
| 1.6 Equivalenza Logica | 8 |
| 1.7 Formalizzazioni in logica proposizionale | 9 |
| 1.8 Teorema di compattezza | 10 |
| 1.8.1 Dimostrazione per i linguaggi numerabili | 12 |
| 1.8.2 Dimostrazione per i linguaggi arbitrari | 13 |
| 1.9 Applicazioni del teorema di compattezza | 14 |
| 1.10 Decidibilità | 15 |
| 1.11 Calcoli deduttivi formali | 17 |
| 1.12 Teorema di completezza | 18 |
| 1.12.1 Dimostrazione del teorema di completezza (WIP) | 18 |
| 2 Logica Predicativa | 19 |
| 2.1 Introduzione | 19 |
| 2.2 Semantica | 21 |
| 2.3 Teorie | 22 |
| 2.4 Isomorfismi tra strutture | 23 |
| 2.5 La teoria DLO | 26 |
| 2.5.1 Numeri razionali | 26 |

| | | |
|-------|--|----|
| 2.6 | DLO: caso generale | 27 |
| 2.6.1 | Sottostrutture, proprietà del testimone | 28 |
| 2.6.2 | Equivalenza tra \mathcal{R} e \mathcal{Q} | 30 |
| 2.7 | Calcolo dei Predicati | 31 |
| 2.7.1 | Proprietà fondamentali del Calcolo dei Predicati | 32 |
| 2.7.2 | Teorema di completezza per la logica predicativa | 32 |
| 2.7.3 | Estensioni di teorie | 33 |

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 $\alpha(A) = \alpha(B)$.

soddisfabilità

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 $T \vDash A_{n+1}$, oppure $A_1, \dots, A_n \vDash A$.

Si ha che:

- $T \not\vDash A$ significa che $\exists \alpha$ t.c. $\alpha(T) = 1 \wedge \alpha(A) = 0$
- $\emptyset \vDash A$ o, equivalentemente $\models A \iff A$ è una tautologia

Lemma 1: Equivalenze

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

sono equivalenti.

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 :

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

dimostrazione

Si dimostra per induzione su n .

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

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

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 (<i>tertium non datur</i>) | $A \vee \neg A \equiv 1$ $A \wedge \neg A \equiv 0$ |
| 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

1.7. Formalizzazioni in logica proposizionale

Il concetto di soddisficiabilità 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) $\neg p_{x,x} \quad \forall x \in X$ (antiriflessività)
- (2) $p_{x,y} \rightarrow \neg p_{y,x} \quad \forall x \in X$ (asimmetria)
- (3) $(p_{x,y} \wedge p_{y,z}) \rightarrow p_{x,z} \quad \forall x, y, z \in X$ (transitività)
- (4) $(p_{x,y} \vee p_{y,x}) \quad \forall x \neq y \in X$ (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.

Colorabilità

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

FINSAT \implies **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:

$$(1) \quad T \models A \iff \exists T_0 \stackrel{\text{fin}}{\subseteq} T \text{ t.c. } T_0 \models A$$

$$(2) \quad T \in \text{SAT} \iff T \in \text{FINSAT}$$

- ① \Rightarrow ②

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

$T \not\models A$ significa $T \cup \{\neg A\} \in \text{SAT}$.

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

- ② \Rightarrow ①

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 ②, 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 $T \in \text{FINSAT}$. \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 $A \in B, C$. Possiamo quindi introdurre $\hat{B} = B \setminus \{A\}$ e $\hat{C} = C \setminus \{A\}$.

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:

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

(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:

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

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 $\{p_{i1}^*, \dots, p_{ik}^*\}$, 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 $p_j \neg \in T^*$. 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 $\alpha(A) = \alpha(B)$.

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 $C = \text{sottoinsieme di } X$ 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 $C = (T_i)$ 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 soddisfabilità - 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 soddisfabilità)
- (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 $\{A, \neg B\} \subseteq T^* \in \text{UNSAT}$, 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:**

$$(1) A = \neg B$$

$$(2) A = B \wedge C$$

$$(3) A = B \vee C$$

$$(4) A = B \rightarrow C$$

TODO

□

1.9. Applicazioni del teorema di compattezza

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 **algoritmicamente 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 algoritmamente 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 algoritmamente, 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*).

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 $\neg(\text{enumerabile} \rightarrow \text{computabilmente enumerabile})$)

Thm. 6

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

- 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. 7

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 $T \models \neg B$.

1.11. Calcoli deduttivi formali

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.:

- $F_k = A$
- $\forall i \in [k]$:
 - 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. 8: **Correttezza**

$$\begin{aligned}\vdash A &\implies \vDash A \\ T \vdash A &\implies T \vDash A\end{aligned}$$

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 $T \vdash A \implies T \vDash A$).

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 .

1.12. Teorema di completezza

Thm. 9: **Teorema di completezza**

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

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

Dal teorema di compattezza sappiamo che

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

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 $\vdash (A_1 \rightarrow (A_2 \rightarrow \dots (A_n \rightarrow A) \dots)) \iff A_1, \dots, A_n \vdash A$.

1.12.1. Dimostrazione del teorema di completezza (WIP)

Per dimostrare il teorema, ci saranno utili le seguenti proprietà:

- (1) $T \vdash A \wedge T \subseteq S \implies S \vdash A$
- (2) $T \vdash A \iff \exists S \stackrel{fin}{\subseteq} T \text{ t.c. } S \vdash A$
- (3) $T \vdash A \wedge (\forall B \in T, S \vdash B) \implies S \vdash A$

2.1. 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 .

2.2. 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 $\mathcal{A} \models G[\alpha]$.
- $\mathcal{A} \models (G \wedge H)[\alpha]$ 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 $\mathcal{A} \models G[\alpha(x/a)]$.

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: Soddisfacibilità, Validità

Se $\exists \alpha \text{ t.c. } \mathcal{A} \models F[\alpha]$, 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.

2.3. 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 $Mod(T) = \{\mathcal{A} : \mathcal{A} \models T\}$

(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 $Conseq(T) = \{E : T \models E\}$

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:

$$\forall v_1 \forall v_2 \forall v_3 ((v_1 \circ v_2) \circ v_3 = v_1 \circ (v_2 \circ v_3)).$$

$$\forall v_1 ((v_1 \circ e = v_1) \wedge (e \circ v_1 = v_1))$$

$$\forall v_1 \exists v_2 ((v_1 \circ v_2 = e) \wedge (v_2 \circ v_1 = e))$$

che formano la teoria dei gruppi T_G .

I modelli della teoria dei gruppi sono le strutture che chiamiamo gruppi, quindi si ha che $Mod(T_G) =$ 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 ” $\{E : G \models E\}$.

Si osserva facilmente che la teoria di una singola struttura \mathcal{M} è sempre completa.

D'altra parte, l'insieme delle conseguenze logiche di una teoria T non è necessariamente completo (nel senso di contenere E o $\neg E$ per ogni possibile enunciato E). Per esempio, le conseguenze degli assiomi di gruppo (ossia la teoria T_{Gruppi}) non formano un insieme completo: 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.

2.4. 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. 10: Isomorfismo tra due strutture

$\mathcal{A} \cong \mathcal{B} \implies \mathcal{A}$ 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:**

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

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

$$(2) \quad t_1 = t_2$$

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

- **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 .

$$\begin{aligned}\mathcal{A} &\models (\exists y\phi)[\vec{a}] \\ \iff \exists c &\in A \text{ tale che } \mathcal{A} \models \phi[\vec{a}, c] \quad (\text{def. di } \exists) \\ \iff \exists c &\in A \text{ tale che } \mathcal{B} \models \phi[h(\vec{a}), h(c)] \quad (\text{ip. ind. su } \phi) \\ \text{Poiché } h &\text{ è suriettiva, l'insieme } \{h(c) \mid c \in A\} \text{ è uguale all'insieme base } B. \text{ Posto } b = h(c): \\ \iff \exists b &\in B \text{ tale che } \mathcal{B} \models \phi[h(\vec{a}), b] \\ \iff \mathcal{B} &\models (\exists y\phi)[h(\vec{a})] \quad (\text{def. di } \exists)\end{aligned}$$

(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

$$\begin{aligned}\mathcal{A} \cong \mathcal{B} &\iff \exists h : A \rightarrow B \text{ t.c.} \\ \forall i &\in I, j \in J, \quad \forall (a_1, \dots, a_t) \in A^t \\ \forall (a_1, \dots, a_t) &\in A^t, \quad (a_1, \dots, a_t) \in R_i^A \iff (h(a_1), \dots, h(a_t)) \in R_i^B \\ h(f_j^A(a_1, \dots, a_t)) &= f_j^B(h(a_1), \dots, h(a_t))\end{aligned}$$

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 ($t = f(t_1, \dots, t_p)$)

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

La proprietà è dimostrata per tutti i termini. □

Corollary for Thm. 10: 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. 11: 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.

dim grafi

2.5. La teoria DLO

2.5.1. 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 $x < y$ 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. 12: \mathbb{Q} e DLO

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

2.5.1.1. Isomorfismo: caso numerabile

Thm. 13: 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 \preceq un ordine totale denso senza estremi su A è isomorfa a \mathbb{Q}, \leq .

È naturale definire l'isomorfismo così: $\mathcal{B} \cong \mathcal{Q} \iff \exists h : \mathbb{Q} \hookrightarrow B$ 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 asseriscono 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. 14: 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 $\mathcal{A} \models$ DLO e $\mathcal{B} \models$ DLO. Allora, $\mathcal{A} \cong \mathcal{B}$.

2.6. DLO: caso generale

Ci chiediamo: data una struttura $\mathcal{D} = (D, \leq_D) \models$ 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 lingaggio 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 $\mathcal{Q} \equiv \mathcal{R}$.

2.6.1. Sottostrutture, proprietà del testimone

Introduciamo come prima cosa il concetto di sottostruttura:

Def. 24: 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. 25: 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. 15: 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. 15: 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. 15: 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 $\mathcal{Q} \prec \mathcal{R}$.

2.6.2. 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.27) 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 soddisficiabilità 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 (assioma della scelta).

Def. 26: 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 \mathcal{F})

Definiamo

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

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

fine

2.7. 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. 27: 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.17)

Gli assiomi predicativi:

- (2) $\forall x F \rightarrow F[x/t]$ con t termine libero per x in F (per “libero” vedi p.20)

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

- (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:

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

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

(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 $\forall x$)

2.7.1. 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)
- $T \vdash A \wedge T \subseteq S \implies S \vdash A$
- 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. 16: Teorema di deduzione per il calcolo predicativo

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

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

Altre proprietà:

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

2.7.2. 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. 28: 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. 29: 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, $T \vdash A \implies T \vDash A$ e $T \vdash \neg A \implies T \vDash \neg A$, e se valgono sia $T \vDash A$ che $T \vDash \neg A$ sappiamo che T è insoddisfacibile ($Mod(T) = \emptyset$).

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

Thm. 17: Equivalenza tra le due forme di teorema di completezza

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

proof

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

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. Questo implica logicamente qualsiasi cosa ($T \vdash A$ e $T \vdash \neg A$). Per (1), quindi, $T \models A$ e $T \models \neg A$, quindi T incoerente (per contrapposizione, otteniamo l'implicazione originale).

2.7.3. Estensioni di teorie

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

Def. 30: Estensione

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

Def. 31: 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 5: 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 $\exists E$ 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.