Logica Matematica

Lorenzo Carlucci

Dispense Sapienza
A.A. 2023/2024

"tutto ciò che non so ma che so come spiegare."

– Lorenzo Carlucci

Indice

LOGICA MATEMATICA

A.A. 23/24, DISPENSA N. 1

SOMMARIO. Sintassi e semantica della logica proposizionale. Tavole di verità.

1. INTRODUZIONE

La Logica Proposizionale si occupa di studiare le proprietà di alcuni costrutti (o peratori) logici utilizzati nel linguaggio naturale e nella pratica scientifica e matematica, quali il *non* (negazione), l'*oppure* (disgiunzione), l'*e* (congiunzione), il *se ... allora* (implicazione) o il *se e solo se* (equivalenza, doppia implicazione).

Esempio 1.1. Consideriamo il seguente semplice argomento aritmetico.

(1) Se $a = 0$ o $b = 0$ allora $a \cdot b = 0$.
(2) $a \cdot b \neq 0$.
(3) $a \neq 0$ e $b \neq 0$.

Intuitivamente la terza proposizione è la conclusione di un argomento che ha come premesse le prime due. Come si formalizza? Per prima cosa si individuano quelle parti che non possono essere ulteriormente analizzate in termini di costrutti logici e che possono essere vere o false (dette parti atomiche). Nel nostro caso, queste parti atomiche sono $a = 0$, $b = 0$, e $a \cdot b = 0$.

Associamo a ciascuna di queste parti atomiche una distinta lettera: a $a = 0$ associamo A, a $b = 0$ associamo B, a $a \cdot b = 0$ associamo C. Infine sostituiamo i costrutti logici del linguaggio naturale (Se ... allora, o, e, non) con dei simboli formali, detti connettivi Booleani: $\neg$ per la negazione, $\vee$ per la disgiunzione, $\wedge$ per la congiunzione, $\rightarrow$ per l'implicazione (se... allora), e $\leftrightarrow$ per la doppia implicazione (se e solo se).

Otteniamo la seguente formalizzazione,

(i) $(A \vee B) \rightarrow C$.
(ii) $(\neg C)$.
(iii) $(\neg A \wedge \neg B)$.

dove, intuitivamente, leggiamo $A \vee B$ come "A oppure B", $\neg C$ come "non C", etc.

Consideriamo ora il seguente argomento verbale.

(a) Se il padre è alto o la madre è alta allora il figlio è alto.
(b) Il figlio è basso.
(c) Il padre è basso e la madre è bassa.

Ovviamente la prima premessa (a) è empiricamente falsa, mentre la prima premessa (1) è matematicamente vera. Ciò nonostante, riconosciamo intuitivamente che il ragionamento è valido (o corretto).

Quando diciamo che l'argomento è valido (o corretto) intendiamo dire che *se* le premesse sono vere, *allora* è vera la conclusione; non che le premesse sono vere.

Se tentiamo una formalizzazione dell'argomento ci rendiamo conto facilmente che otteniamo la stessa formalizzazione che abbiamo ottenuto per l'argomento precedente, dove scriviamo A per "il padre è alto", B per "la madre è alta" e C per "il figlio è alto". I due argomenti risultano pertanto identici. La logica formale permette di individuare la *forma logica* comune ad argomenti che trattano di oggetti e strutture diverse, come è il caso negli esempi qui sopra.

La logica si occupa della validità di ragionamenti *indipendentemente dal significato delle loro componenti*. In questo è parente dell'Algebra moderna.

Dispense preparate da Lorenzo Carlucci; segnalare refusi o errori a lorenzo.carlucci@uniroma1.it.

> Quanti sono a conoscenza dello stato attuale della teoria dell'algebra simbolica sono consapevoli del fatto che la validità dei processi di analisi non dipende dall'interpretazione dei simboli impiegati, ma soltanto dalle leggi della loro combinazione. Ogni sistema di interpretazione che non intacchi la verità delle relazioni presupposte è egualmente ammissibile. (George Boole, *L'analisi matematica della logica*, 1847)

2. Linguaggio e proposizioni formali

Il primo passo è quello di definire un linguaggio formale completamente rigoroso e passibile di uno studio matematico. Per la Logica Proposizionale la procedura è semplice: le proposizioni sono formalizzate da formule ottenute applicando i connettivi Booleani ad alcuni mattoncini atomici, detti *variabili proposizionali*. Queste variabili rappresentano, intutivamente, proposizioni non ulteriormente analizzabili (relativamente al contesto di interesse) in termini di operatori logici come negazione, congiunzione, disgiunzione, implicazione o doppia implicazione.

Una scelta di tali variabili definisce un cosidetto *linguaggio proposizionale*.

Definizione 2.1 (Linguaggio proposizionale)**.** Un *linguaggio proposizionale* è un insieme infinito $\mathcal{L}$ di simboli detti variabili proposizionali, tipicamente denotato come $\{p_i : i \in I\}$ dove I è un insieme detto di *indici*.

In alcuni casi un linguaggio con un numero finito di variabili è più che sufficiente ma per convenienza tecnica e uniformità abbiamo richiesto che un linguaggio proposizionale si componga di infinite variabili.

La scelta del linguaggio dipende dalla natura del problema o ragionamento (matematico o verbale) che vogliamo formalizzare. L'idea è quella di avere a disposizione una variabile proposizionale per ogni singolo blocco elementare di informazioni con cui abbiamo a che fare.

Una volta fissati i simboli corrispondenti ai fatti elementari passiamo a considerare la formalizzazione del concetto intuitivo di proposizione. L'idea è di considerare proposizione (o formula) tutto ciò che posso ottenere applicando connettivi Booleani a variabili proposizionali.

Definizione 2.2 (Proposizioni/Formule)**.** Sia $\mathcal{L}$ un linguaggio proposizionale. L'insieme delle *proposizioni* (o *formule ben formate*) in $\mathcal{L}$ è il minimo insieme X di stringhe finite di simboli in $\mathcal{L} \cup \{\neg, \vee, \wedge, \rightarrow, \leftrightarrow, (,)\}$ tale che

(1) Tutti gli elementi di $\mathcal{L}$ (le variabili proposizionali) sono in X, e
(2) Se A è in X allora $(\neg A)$ è in X, e
(3) Se A e B sono in X allora $(A \wedge B)$, $(A \vee B)$, $(A \rightarrow B)$ sono in X.

Denotiamo con $\mathrm{PROP}_{\mathcal{L}}$ (o $\mathsf{FML}_{\mathcal{L}}$) l'insieme delle proposizioni (o formule) nel linguaggio $\mathcal{L}$. Se $\mathcal{L}$ è irrilevanto o chiaro dal contesto, scriviamo PROP (o FML).

Osservazione 2.3. Per facilitare la leggibilità delle formule stabiliamo le seguenti regole di precedenza: $\neg$ ha precedenza su $\vee, \wedge$ e questi ultimi hanno precedenza su $\rightarrow$ e $\leftrightarrow$.

Cosa significa nella Definizione precedente che PROP è *il minimo insieme tale che...*? Quello che si intende è che, se Y è un qualunque insieme che soddisfa (1)(2)(3), allora $\mathrm{PROP} \subseteq Y$. Come sappiamo che un tale Y esiste? Possiamo argomentare come segue.

Sia $\mathcal{L}$ un linguaggio proposizionale. Definiamo una famiglia numerabile di insiemi di stringhe, per ricorsione su n.

$$\mathbf{F}_0 := \mathcal{L}$$
$$\mathbf{F}_{n+1} := \mathbf{F}_n \cup \{(\neg A) \ : \ A \in \mathbf{F}_n\} \cup \{(A \wedge B), (A \vee B), (A \rightarrow B), \ : \ A, B \in \mathbf{F}_n\}$$
$$\mathbf{F} := \bigcup_{n \in \mathbf{N}} \mathbf{F}_n$$

Si può dimostrare che $\mathbf{F}$ coincide con l'insieme delle proposizioni nel linguaggio $\mathcal{L}$.

Proposizione 2.4. *L'insieme delle proposizioni coincide con* $\mathbf{F}$.

Dimostrazione. Cominciamo col dimostrare che $\mathbf{F}$ soddisfa le clausole che definiscono PROP. Ovviamente tutte le variabili proposizionali sono in $\mathbf{F}$ perché sono in $\mathbf{F}_0$. Se A e B appartengono a $\mathbf{F}$ allora $A \in \mathbf{F}_n$ e $B \in \mathbf{F}_m$ per qualche n, m. Assumiamo, senza perdita di generalità, che $n \leq m$. Allora $A, B \in \mathbf{F}_m$ poiché $\mathbf{F}_n \subseteq \mathbf{F}_m$ e dunque $(\neg A), (A \square B) \in \mathbf{F}_{m+1}$.

Sia $\mathcal{G}$ una famiglia che soddisfa le clausole definitorie di PROP. Allora $\mathbf{F}_0 \subseteq \mathcal{G}$ poiché $\mathbf{F}_0 = \mathcal{L}$. Se $\mathbf{F}_n \subseteq \mathcal{G}$ allora $\mathbf{F}_{n+1} \subseteq \mathcal{G}$ per le proprietà di chiusura di $\mathcal{G}$ e la definizione di $\mathbf{F}_{n+1}$. Dunque $\mathbf{F} \subseteq \mathcal{G}$ e abbiamo dimostrato che $\mathbf{F}$ è il più piccolo insieme che soddisfa le clausole definitorie di PROP. □

L'identità i PROP con $\mathbf{F}$ permette di associare a ogni proposizione una misura della sua complessità sintattica: se $A \in \text{PROP}$, dato che $\text{PROP} = \mathbf{F} = \bigcup_{n\in\mathbf{N}} \mathbf{F}_n$, esiste un minimo n tale che $A \in \mathbf{F}_n$. Questo n (che potremmo chiamare *rango* o *grado* di A) conta il numero di connettivi booleani presenti in A.

3. Semantica della Logica Proposizionale: assegnamenti

Finora abbiamo introdotto un linguaggio formale e simbolico – si tratta di un oggetto puramente sintattico. Uno dei nostri obiettivi è di studiare rigorosamente la nozione di argomento o ragionamento valido. Abbiamo in mente argomenti di tipo *ipotetico-deduttivo* quali si incontrano in quasi tutte le aree del sapere umano: Se valgono le premesse (o ipotesi) $A_1, \ldots, A_n$ allora vale la conclusione A. Altrimenti detto: assumendo che siano vere le premesse, segue necessariamente che è vera la conclusione. Risulta dunque naturale definire rigorosamente il concetto di *verità* di una proposizione nel nostro formalismo.

Questo è il compito della cosidetta semantica – ma si noti bene che il termine semantica, che di solito rimanda a una nozione di significato, in questo caso ha una accezione molto ristretta: in Logica Proposizionale il significato di una proposizione si riduce alla sua verità o falsità.

In questo corso studiamo la cosidetta *logica classica* che è caratterizzata dall'assumere che ogni proposizione può avere uno solo tra due possibili valori di verità, ossia essere *vera* oppure *falsa*. Esistono altre logiche, di interesse in settori specifici, in cui i valori di verità possibili sono più di due!

L'altra idea fondamentale è che la verità (o falsità) di una proposizione complessa (ossia contenente qualche connettivo logico) viene definita esclusivamente in funzione della verità (o falsità) delle sottoformule immediate che la compongono.

L'idea è piuttosto naturale: siamo inclini a a considerare vera una proposizione del tipo "$1 < 5$ e π è irrazionale" se e soltanto se sono veri entrambi i congiunti. Questa regola vale indipendentemente da quali sono i congiunti. La applicheremmo in generale a qualsiasi proposizione che abbia la forma di una congiunzione di due proposizioni: A e B, per esempio "Dio esiste e la vita è bella"; o "Se il gas G ha bassa concentrazione allora il gas G ha anche bassa pressione e se il gas G ha bassa pressione allora il gas G si comporta come un gas ideale", o "Se $f(x)$ è continua in $[a, b]$ e derivabile in (a, b) allora esiste $\xi \in (a, b)$ tale che $\frac{f(b)-f(a)}{b-a} = f'(\xi)$" (Teorema della Media).

Analogamente siamo inclini a giudicare vera (risp. falsa) una proposizione negativa, del tipo: "π non è algebrico" oppure "Dio non esiste" se e soltanto se la proposizione negata ("π è algebrico", "esiste Dio") è falsa (risp. vera), indipendentemente da quale essa sia.

Queste considerazioni ci portano all'idea di valutare la verità o falsità di una proposizione complessa come *funzione* della verità o falsità delle sue componenti immediate.

Se iteriamo il processo a un certo punto ci imbatteremo in proposizioni non ulteriormente analizzabili. Per esempio: "il gas G ha bassa pressione" o "il gas G si comporta come un gas ideale", oppure "π è irrazionale", "π è algebrico". La verità o falsità di queste componenti atomiche dipende dal significato dei termini che vi compaiono e dallo stato di cose nell'universo di riferimento. Ciò nonostante, per valutare la verità o falsità delle proposizioni composte che le contengono, ci interessa soltanto sapere se queste componenti atomiche sono vere o false per poter calcolare a ritroso il valore di verità della proposizione complessa di partenza.

Formalmente procediamo come segue: definiamo la nozione di assegnamento (o valutazione) come associazione di un valore di verità (vero/falso o 1/0) a tutte le variabili proposizionali del linguaggio. In base a un tale assegnamento definiamo ricorsivamente il valore di verità di una qualunque proposizione arbitrariamente complessa come funzione dei valori di verità delle sue componenti. La verità o falsità di una proposizione complessa, in generale, dipenderà dunque dall'assegnamento usato per dare un valore di verità alle variabili proposizionali.

Definizione 3.1 (Assegnamento)**.** Un *assegnamento* è una funzione di tipo

$$\alpha : \mathcal{L} \to \{0, 1\}.$$

I numeri $0, 1$ vengono detti *valori di verità*, e sono intuitivamente da identificarsi come *Falso* e *Vero*.

Vogliamo estendere un qualunque assegnamento $\alpha : \mathcal{L} \to \{1, 0\}$ a una funzione

$$\alpha' : \mathrm{PROP}_{\mathcal{L}} \to \{0, 1\}.$$

Lo facciamo dando delle regole per calcolare ricorsivamente il valore di α' su una proposizione A come funzione dei valori di α' sulle sottoformule immediate di A. Per alleggerire la notazione, a rischio di ambiguità, usiamo α per indicare la funzione di tipo $\mathrm{PROP} \to \{0, 1\}$ ottenuta estendendo v secondo le regole seguenti.

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

$$\alpha((A \vee B)) := \begin{cases} 0 & \text{se } \alpha(A) = \alpha(B) = 0 \\ 1 & \text{altrimenti.} \end{cases}$$

$$\alpha((A \wedge B)) := \begin{cases} 1 & \text{se } \alpha(A) = \alpha(B) = 1 \\ 0 & \text{altrimenti.} \end{cases}$$

$$\alpha((A \to B)) := \begin{cases} 0 & \text{se } \alpha(A) = 1 \text{ e } \alpha(B) = 0 \\ 1 & \text{altrimenti.} \end{cases}$$

Osservazione 3.2. La definizione del valore di verità di una implicazione $A \to B$ data sopra definisce la cosidetta *implicazione materiale*. Secondo questa definizione una implicazione $A \to B$ è vera nei tre casi seguenti.

(1) A e B sono vere,
(2) A è falsa e B è vera,
(3) A è falsa e B è falsa.

In breve, $A \to B$ è falsa se e solo se A è vera e B è falsa.

Si noti che dalla definizione segue che una implicazione può essere vera senza che vi sia connessione causale o di significato tra antecedente e conseguente. Per esempio: "Se tutti i quadrati sono pari allora π è irrazionale" risulta vera secondo questa definizione.

In secondo luogo segue dalla definizione cha una implicazione è sempre vera se il suo antecedente è falso. Questo rispecchia la pratica matematica di considerare vera a vuoto una proposizione ipotetica la cui premessa non si applica. Per esempio, la definizione di "X è sottinsieme di Y" prevede che comunque io scelga un elemento x, se x è in X allora deve essere anche in Y. L'esistenza di elementi non in X ma in Y non falsifica la definizione, così come l'esistenza di elementi né in X né in Y. Analogamente: "Se H e K sono sottogruppi normali del gruppo G allora HK è un sottogruppo normale di G" non viene falsificata dall'esistenza di sottogruppi non normali X, Y di G tali che XY non è un sottogruppo normale.

Ancora, in Matematica consideriamo vere implicazioni con premesse false, del tipo: "Se π è razionale allora 2π è razionale".

La definizione della semantica dell'implicazione può anche giustificarsi considerando le *promesse*: "Se arrivi in orario al colloquio allora otterrai il lavoro", "Se credi in Dio allora avrai la vita eterna", "Se non ti ubriachi allora non verrai aggredito", etc. È naturale (ossia coerente con l'uso comune) dire che una promessa di questo tipo non viene mantenuta nel caso seguente: viene verificata la condizione ma non viene verificata la conseguenza (e.g. Arrivo in orario al colloquio ma poi non ottengo il lavoro, etc.).

La scelta della definizione di $v(A \to B)$ in funzione di $v(A)$ e $v(B)$, e in particolare i punti (2) e (3), possono giustificarsi ulteriormente come segue.

Nel nostro sistema vogliamo che la proposizione $(A \wedge B) \to B$ sia sempre vera, qualunque siano A e B. Vediamo come questa richiesta impone un vincolo alla definizione di $v(A \to B)$ che spiega i casi poco intuitivi quali $0 \to 1$, $1 \to 1$ e $0 \to 0$.

Il caso $1 \to 1$ deve essere vero perché corrisponde al caso in cui $(A \wedge B)$ è vera e dunque sia A che B sono vere. Il caso $0 \to 0$ deve essere vero perché corrisponde al caso in cui $(A \wedge B)$ è falso perché B è falso. Il caso $0 \to 1$ 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 $1 \to 0$, che non corrisponde a nessun caso di $(A \wedge B) \to B$ (né di $(A \wedge B) \to A$).

Un'altra giustificazione (parziale) della scelta della definizione della tavola di verità di $\to$ è che vogliamo che valga l'implicazione seguente, che formalizza il ragionamento per contrapposizione:

$$(A \to B) \to (\neg B \to \neg A).$$

Che vincoli impone questa richiesta alla tavola di verità di $\to$?

Se A e B sono vere la premessa $(A \to B)$ è vera ma il conseguente $(\neg B \to \neg A)$ è di tipo $0 \to 0$.

4. Tavole di verità

Osserviamo che è possibile presentare i casi della definizione di v qui sopra in modo compatto usando le cosidette Tavole di Verità, ossia una presentazione tabulare della funzione di valutazione sopra definita.

Per esempio, possiamo riscrivere la definizione di $\alpha((\neg A))$ in forma tabulare come segue.

A	$\neg A$
1	0
0	1

Analogamente possiamo riscrivere la definizione di $v((A \vee B))$ in forma tabulare come segue.

A	B	$(A \vee B)$
1	1	1
1	0	1
0	1	1
0	0	0

Lo stesso possiamo fare per tutti gli altri casi.

La possibilità di organizzare in una tabella i valori di verità di una proposizione composta come funzione dei valori di verità delle sue componenti sopra accennato può essere generalizzato a proposizioni qualunque.

Data una proposizione A che contiene le variabili proposizionali $p_1, \ldots, p_n$ distinte e le sottoformule $B_1, \ldots, B_k$, possiamo organizzare la Tavola di verità di A come segue. Nelle prime n colonne scriviamo tutti i possibili valori assunti dalle variabili proposizionali $p_1, \ldots, p_n$. Nelle restanti colonne scriviamo i valori assunti dalle sottoformule di A in ordine crescente di complessità (misurata in termini di rango).

Esempio Sia $A = ((P \vee Q) \to (R \vee (R \to Q)))$.

P	Q	R	$(R \to Q)$	$(R \vee (R \to Q))$	$(P \vee Q)$	A
1	1	1	1	1	1	1
1	1	0	1	1	1	1
1	0	1	0	1	1	1
1	0	0	1	1	1	1
0	1	1	1	1	1	1
0	1	0	1	1	1	1
0	0	1	0	1	0	1
0	0	0	1	1	0	1

Esempio Sia $A = ((\neg P) \wedge Q) \to R)$.

P	Q	R	$(\neg P)$	$((\neg P) \wedge Q)$	A
1	1	1	0	0	1
1	1	0	0	0	1
1	0	1	0	0	1
1	0	0	0	0	1
0	1	1	1	1	1
0	1	0	1	1	0
0	0	1	1	0	1
0	0	0	1	0	1

Possiamo costruire (meccanicamente) la tavola di verità di una qualunque proposizione A. Se la proposizione contiene n variabili proposizionali, la sua tavola di verità ha 2^n righe. Ogni assegnamento di valori di verità alle variabili proposizionali di A corrisponde a una riga della tavola di verità di A, e viceversa.

5. Completezza Funzionale

Possiamo chiederci se il formalismo introdotto sia sufficientemente espressivo. Siamo in grado di esprimere tutti i concetti di natura logica che incontriamo nel linguaggio naturale?

Solo se. Consideriamo per esempio la locuzione "solo se", che possiamo incontrare in contesti del tipo: "Un numero è divisibile per 4 *solo se* è pari", "Giorgio ha dato l'esame di Algebra 2 *solo se* ha superato l'esame di Algebra 1", "Sei iscritto al corso di Logica *solo se* sei iscritto all'Università", "Oggi è Pasqua *solo se* domani è Lunedì" "Sei cittadino italiano *solo se* sei cittadino europeo".

Analizziamo le condizioni di verità che naturalmente associamo a queste locuzioni: per es. "Sei cittadino italiano solo se sei cittadino europeo" è usualmente ritenuta equivalente a "Se non sei cittadino europeo allora non sei cittadino italiano". Questo ci suggerisce immediatamente di esprimere proposizioni del tipi "A solo se B" come "Se non B allora non A"; in formule $\neg B \to \neg A$. Non abbiamo perciò bisogno di introdurre un simbolo speciale per il costrutto "solo se". Inoltre possiamo osservare facilmente (e verificare con una tavola di verità) che $\neg B \to \neg A$ si comporta esattamente come $A \to B$: associa i medesimi valori di verità ai medesimi assegnamenti di verità alle sue componenti A e B. Diremo in questo caso che le due proposizioni sono logicamente equivalenti. Possiamo dunque formalizzare proposizioni del tipo "A solo se B" come $A \to B$ ("Se A allora B").

Disgiunzione esclusiva. In alcuni contesti di uso naturale del linguaggio la disgiunzione (oppure) viene intesa come esclusiva, ossia si esclude che i due termini possano essere veri simultaneamente. Per esempio "O se fascista o sei comunista", "O sei maschio o sei femmina", "O sei cattolico o sei protestante"; anche se i due disgiunti non sono immediatamente l'uno la negazione dell'altro. Le condizioni di verità naturalmente associate a proposizioni di questo tipo (che chiameremo disgiunzioni esclusive) sono le seguenti: almeno uno dei disgiunti è vero ma non entrambi. La tavola di verità di questa interpretazione è dunque la seguente:

A	B	?
1	1	0
1	0	1
0	1	1
0	0	0

La domanda è se possiamo scrivere una proposizione formale che abbia esattamente questa tavola di verità. Otteniamo una tale proposizione semplicemente parafrasando le condizioni di verità intuitive sopra descritte (almeno uno dei disgiunti è vero ma non entrambi): A è vero e B è falso o A è falso e B è vero. In simboli $(A \wedge \neg B) \vee (\neg A \wedge B)$. Si verifica facilmente che la tavola di verità di questa proposizione è esattamente quella qui sopra.

Se e solo se. Un altro costrutto naturale è il "se e solo se" (molto frequente in Matematica). Le condizioni di verità naturalmente associate a una proposizione del tipo "A se e solo se B" sono: se A è vero allora B è vero e se B è vero allora A è vero. In forma di tavola di verità:

A	B	?
1	1	1
1	0	0
0	1	0
0	0	1

Si vede facilmente che la proposizione $(A \to B) \wedge (B \to A)$ ha esattamente la tavola di verità qui sopra. Abbiamo tradotto in simboli il significato naturale del "se e solo se", scomponendolo in un "se ... allora" e in un "... solo se ..." (che sappiami già formalizzare). In alternativa possiamo esprimere le condizioni di verità intuitive di "A se e solo se B" come segue: A è vero e B è vero oppure A è falso e B è falso. Questo ci porta alla proposizione formale $(A \wedge B) \vee (\neg A \wedge \neg B)$. Si può verificare facilmente che la tavola di verità di questa proposizione è quella di sopra. Possiamo concludere che $(A \to B) \wedge (B \to A)$ e $(A \wedge B) \vee (\neg A \wedge \neg B)$ sono logicamente equivalenti.

Le osservazioni qui sopra ci conducono alla domanda generale: data una tavola di verità arbitraria con n argomenti (e 2^n righe), esiste una proposizione A (con n variabili proposizionali) che ha esattamente quella tavola di verità?

LOGICA MATEMATICA

A.A. 23/24, DISPENSA N. 2

SOMMARIO. Completezza funzionale. Forme Normali.

1. Completezza funzionale

Il prolema generale. Una proposizione A contenente soltanto le n variabili proposizionali $\{p_1, \dots, p_n\}$ determina una funzione di n argomenti, $f_A : \{0,1\}^n \to \{0,1\}$ definita come segue: Per determinare il valore di f_A su un argomento $(x_1, \dots, x_n) \in \{0,1\}^n$ considero *un arbitrario assegnamento* α tale che $\alpha(p_k) = x_k$ per $k \in [1,n]$ e pongo $f_A(x_1, \dots, x_n) = \alpha(A)$. Per assicurasi che la definizione è ben posta basta osservare che se α e β sono due assegnamenti che coincidono sulle n variabili proposizionali che compaiono in A, allora abbiamo $\alpha(A) = \beta(A)$.

Chiamiamo una funzione di tipo $\{0,1\}^n \to \{0,1\}$ una *funzione di verità* o *funzione booleana*.

Risulta naturale chiedersi se con i connettivi che abbiamo scelto siamo capaci di rappresentare il comportamento di qualunque funzione di verità

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

a n argomenti, per $n \in \mathbf{N}$?

Teorema 1.1. *Sia $f : \{0,1\}^n \to \{0,1\}$ una funzione di verità. Esiste una proposizione A contenente n variabili proposizionali $\{p_1, \dots, p_n\}$ tale che per ogni assegnamento α,*

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

Dimostrazione. Per induzione su n.

Se $n = 1$ abbiamo solo quattro possibili f.

$$f_1(0) = 0, f_1(1) = 0$$

$$f_2(0) = 1, f_2(1) = 1$$

$$f_3(0) = 0, f_3(1) = 1$$

$$f_4(0) = 1, f_4(1) = 0$$

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

Se $n > 1$, scriviamo il grafico di $f : \{0,1\}^n \to \{0,1\}$ in forma di tavola di verità, come segue.

p_1	p_2	$\dots$	p_n	$f(p_1, \dots, p_n)$
0	$\dots$	$\dots$	$\dots$	$\dots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
0	$\dots$	$\dots$	$\dots$	$\dots$
1	$\dots$	$\dots$	$\dots$	$\dots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
1	$\dots$	$\dots$	$\dots$	$\dots$

Dispense preparate da Lorenzo Carlucci; segnalare refusi o errori a lorenzo.carlucci@uniroma1.it.

La parte superiore della tabella, senza considerare la prima colonna, definisce una funzione f_0 di $n-1$ argomenti, il cui comportamento è definito dalle prime 2^{n-1} righe. La parte inferiore della tabella, senza considerare la prima colonna, definisce una funzione f_1 di $n-1$ argomenti, il cui comportamento è definito dalle ultime 2^{n-1} righe.

Per ipotesi induttiva sulle funzioni f_0 e f_1, esistono formule B_0 e B_1 con $n-1$ variabili proposizionali (siano senza pregiudizio di generalità $p_2, \ldots, p_n$) tali che, per ogni assegnamento α,

$$\alpha(B_0) = f_0(\alpha(p_2), \ldots, \alpha(p_n)),$$

e

$$\alpha(B_1) = f_1(\alpha(p_2), \ldots, \alpha(p_n)).$$

Vogliamo ora combinare le proposizioni B_0 e B_1 in una nuova proposizione A contenente una variabile proposizionale aggiuntiva e tale che la tavola di verità di A corrisponda alla funzione f. Come fare?

Sia A la formula seguente

$$(\neg p_1 \to B_0) \wedge (p_1 \to B_1).$$

Dimostriamo che A soddisfa la tesi del teorema rispetto alla funzione f. Sia α un assegnamento qualunque. Dimostriamo che

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

Distinguamo due casi.

Se $\alpha(p_1) = 1$, allora $\alpha(\neg p_1 \to B_0) = 1$ e vale

$$\alpha((\neg p_1 \to B_0) \wedge (p_1 \to B_1)) = 1 \text{ se e solo se } \alpha(p_1 \to B_1) = 1.$$

Inoltre, $\alpha(p_1) = 1$, e dunque

$$\alpha(p_1 \to B_1) = 1 \text{ se e solo se } \alpha(B_1) = 1.$$

Ma per quanto visto circa B_1, vale

$$\alpha(B_1) = f_1(\alpha(p_2), \ldots, \alpha(p_n)),$$

e in questo caso – dato che $\alpha(p_1) = 1$ – vale

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

Dunque in questo caso

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

Se $\alpha(p_1) = 0$, allora $\alpha(\neg p_1) = 1$ e dunque $\alpha(p_1 \to B_1) = 1$. Allora

$$\alpha((\neg p_1 \to B_0) \wedge (p_1 \to B_1)) = 1 \text{ se e solo se } \alpha(\neg p_1 \to B_0) = 1.$$

Inoltre, dato che $\alpha(p_1) = 0$,

$$\alpha((\neg p_1 \to B_0)) = 1 \text{ se e solo se } \alpha(B_0) = 1.$$

Ma per quanto visto circa B_0, vale

$$\alpha(B_0) = f_0(\alpha(p_2), \ldots, \alpha(p_n)),$$

e in questo caso – dato che $\alpha(p_1) = 0$ – vale

$$f(\alpha(p_1), \alpha(p_2), \ldots, \alpha(p_n)) = f(0, \alpha(p_2), \ldots, \alpha(p_n)) = f_0(\alpha(p_2), \ldots, \alpha(p_n)).$$

Dunque in questo caso

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

□

Esercizio Dimostrare il Teorema combinando B_0 e B_1 in una formula senza usare $\to$.

2. Forme Normali

Come accade in Algebra, anche in Logica è possibile – e utile – individuare alcune forme normali o canoniche.

Chiamiamo "letterale" una variabile proposizionale o una negazione di una variabile proposizionale.

Diciamo che A è in Forma Normale Disgiuntiva (DNF) se A è una disgiunzione di congiunzioni di letterali, ossia è della forma seguente, dove gli $A_{i,j}$ sono letterali.

$$(A_{1,1} \wedge A_{1,2} \wedge \cdots \wedge A_{1,m_1}) \vee (A_{2,1} \wedge A_{2,2} \wedge \cdots \wedge A_{2,m_2}) \cdots \vee (A_{n,1} \wedge A_{n,2} \wedge \cdots \wedge A_{n,m_n}).$$

Teorema 2.1 (Forma Normale Disgiuntiva)**.** *Per ogni A esiste A^{DNF} tale che A^{DNF} è una DNF che rappresenta la stessa funzione booleana di A*

Dimostrazione. Scriviamo la tavola di verità di A

p_1	p_2	$\dots$	p_n	A
0	$\dots$	$\dots$	$\dots$	$\dots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
0	$\dots$	$\dots$	$\dots$	$\dots$
1	$\dots$	$\dots$	$\dots$	$\dots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
1	$\dots$	$\dots$	$\dots$	$\dots$

La tavola ha 2^n righe. Per ogni $1 \leq i \leq n$, la riga i determina un assegnamento di valori di verità a $p_1, \dots, p_n, A$, che chiamiamo α_i. La riga i-esima dice che se p_1 ha valore $\alpha_i(p_1)$, e p_2 ha valore $\alpha_i(p_2)$, ..., e p_n ha valore $\alpha_i(p_n)$ allora A ha il valore $\alpha_i(A)$. I casi in cui A è vera sono completamente descritti dalle righe i in cui $v_i(A) = 1$. In altre parole, A è vera se e solo se le variabili proposizionali $p_1, \dots, p_n$ assumono i valori $\alpha_i(p_1), \dots, v_i(p_n)$ per una qualche riga i tale che $\alpha_i(A)$. Dunque, per ogni assegnamento α, $\alpha(A) = 1$ se e solo se α coincide con α_i su $p_1, \dots, p_n$ dove i è una riga in cui A è vera.

Usiamo i letterali per rappresentare all'interno del linguaggio i due casi $\alpha_i(p_{i,j}) = 1$ e $\alpha_i(p_{i,j}) = 0$, per $i \in \{1, \dots, 2^n\}$ e $j \in \{1, \dots, n\}$. Definiamo $p'_{i,j} = p_{i,j}$ se $\alpha_i(p_{i,j}) = 1$ e $p'_{i,j} = \neg p_{i,j}$ se $\alpha_i(p_{i,j}) = 0$. Consideriamo ora l'insieme $I_1 \subseteq \{1, \dots, 2^n\}$ delle righe in cui A ha valore 1. Per $i \in I_1$, alla riga i-esima associamo la congiunzione

$$p'_{i,1} \wedge \cdots \wedge p'_{i,n}.$$

Questa congiunzione rappresenta l'assegnamento α_i, nel senso che, per ogni assegnamento α,

$$\alpha(p'_{i,1} \wedge \cdots \wedge p'_{i,n}) = 1 \Rightarrow \alpha(A) = 1.$$

Infatti vale che

$$\alpha(p'_{i,1} \wedge \cdots \wedge p'_{i,n}) = 1 \Rightarrow \alpha(p'_{i,1}) = \alpha_i(p_{i,1}), \dots, \alpha(p'_{i,n}) = \alpha_i(p_{i,n}).$$

Dato che per ogni α vale $\alpha(A) = 1$ se e solo se per qualche $i \in I_1$, α coincide con α_i sulle variabili $p_1, \dots, p_n$ di A, abbiamo che l'intera tavola di verità di A è rappresentata dalla DNF

$$\bigvee_{i \in I_1} \bigwedge_{j \in \{1,\dots,n\}} p'_{i,j}.$$

□

Usiamo $\bigwedge_{i \leq n} A_i$ come abbreviazione di

$$A_1 \wedge A_2 \wedge \cdots \wedge A_n.$$

e analogamente $\bigvee_{i \leq n} A_i$ come abbreviazione di

$$A_1 \vee A_2 \vee \cdots \vee A_n.$$

Con questa notazione, A è una CNF se è della forma

$$\bigwedge_{i \leq n} \bigvee_{j \leq m_i} A_{i,j},$$

ed è in DNF se è della forma

$$\bigvee_{i \leq n} \bigwedge_{j \leq m_i} A_{i,j},$$

dove gli $A_{i,j}$ sono letterali.

Possiamo introdurre una forma duale della DNF invertendo $\vee$ e $\wedge$.

Diciamo che A è in Forma Normale Congiuntiva (CNF) se A è una congiunzione di disgiunzioni di letterali, ossia è della forma seguente, dove gli $A_{i,j}$ sono letterali.

$$(A_{1,1} \vee A_{1,2} \vee \cdots \vee A_{1,m_1}) \wedge (A_{2,1} \vee A_{2,2} \vee \cdots \vee A_{2,m_2}) \wedge \cdots \wedge (A_{n,1} \vee A_{n,2} \vee \cdots \vee A_{n,m_n})$$

Per Esercizio, sviluppare i dettagli di un argomento analogo per ottenere una CNF equivalente a A.

LOGICA MATEMATICA

A.A. 23/24, DISPENSA N. 3

SOMMARIO. Soddisfacibilità e conseguenza logica. Tautologie. Prime proprietà.

1. Soddisfacibilità, conseguenza logica, validità logica

Definizione 1.1 (Proposizioni Soddisfacibili, Tautologie, Contraddizioni)**.** Se A è una proposizione e α un assegnamento tale che $\alpha(A) = 1$ diciamo che α soddisfa A.

Una proposizione A è *soddisfacibile* se esiste un assegnamento α che soddisfa A. Indichiamo con SAT l'insieme delle formule soddisfacibili.

Una proposizione A è una *verità logica* se ogni assegnamento soddisfa A. Si dice anche che A è *valida*, o è una *tautologia*. Indichiamo con TAUT l'insieme delle tautologie.

Una proposizione A è *insoddisfacibile* se nessun assegnamento soddisfa A. Si dice anche che A è *contraddittoria*, o è una *contraddizione*. Indichiamo con UNSAT l'insieme delle proposizioni insoddisfacibili.

Esempio 1.2. Le tautologie possono essere molto semplici e banali, come per esempio $(A \vee \neg A)$ o $(A \wedge B) \to (A \vee B)$, o più complesse, come, per esempio $(A \to B) \to ((A \vee C) \to (B \vee C))$; o decisamente complicate, come per esempio

$$\neg((p_{1,1} \vee p_{1,2}) \wedge (p_{2,1} \vee p_{2,2}) \wedge (p_{3,1} \vee p_{3,2}) \wedge$$
$$(\neg p_{1,1} \vee \neg p_{2,1}) \wedge (\neg p_{1,1} \vee \neg p_{3,1}) \wedge (\neg p_{2,1} \vee \neg p_{3,1}) \wedge (\neg p_{1,2} \vee \neg p_{2,2}) \wedge (\neg p_{1,2} \vee \neg p_{3,2}) \wedge (\neg p_{2,2} \vee \neg p_{3,2}))$$

che (come vedremo) esprime una versione del Principio dei Cassetti o della Piccionaia. Assegnamo alla variabile $p_{i,j}$ il significato intuitivo *il piccione i-esimo va nella piccionaia j-esima*. Con questa interpretazione la formula può parafrasarsi così: è impossibile collocare 3 piccioni in 2 piccionaie senza collocare due piccioni nella stessa piccionaia.

Analogamente le contraddizioni possono essere semplici contraddizioni in termini, quali $p \wedge \neg p$, o proposizioni insoddisfacibili più complesse quali, per es., la negazione della proposizione precedente che formalizza il Principio della Piccionaia per 3 piccioni e 2 piccionaie.

Si osserva che $A \in \text{SAT}$ è un concetto *esistenziale*:

$$A \in \text{SAT} \Leftrightarrow \exists v(v(A) = 1),$$

mentre $A \in \text{TAUT}$ è un concetto *universale*:

$$A \in \text{TAUT} \Leftrightarrow \forall v(v(A) = 1),$$

Esiste la seguente dualità tra TAUT e UNSAT:

$$A \in \text{TAUT} \Leftrightarrow \neg A \in \text{UNSAT}.$$

D'altra parte è ovvio che esistono proposizioni tali che sia $A \in \text{SAT}$ che $\neg A \in \text{SAT}$.

Definizione 1.3 (Teoria)**.** Una teoria (proposizionale) è un insieme di formule (proposizionali).

Si noti che T può essere finita, infinita o vuota.

Definizione 1.4 (Soddisfacibilità)**.** Un assegnamento α *soddisfa* una teoria T se per ogni proposizione A in T si ha $\alpha(A) = 1$. In questo caso diciamo che α è un modello di T e talvolta scriviamo $\alpha \models T$.

Nel caso in cui $T = \{A\}$ diciamo che A è *soddisfacibile*. Se A non è soddisfacibile diciamo che A è insoddisfacibile. Indichiamo con SAT l'insieme delle proposizioni soddisfacibili (*satisfiable*) e con UNSAT l'insieme delle proposizioni insoddisfacibili.

Note preparate da Lorenzo Carlucci, lorenzo.carlucci@uniroma1.it.

Si noti che con queste definizioni la teoria vuota $T = \emptyset$ è soddisfacibile: per ogni assegnamento α, per ogni proposizione A, se $A \in T$ allora $\alpha(A) = 1$.

Le definizioni precedenti possono applicarsi a singole formule A considerando la teoria T formata dalla singola proposizione A. Possiamo così dire che la proposizione A è soddisfacibile se esiste un assegnamento che la mette a 1; che A è una tautologia se ogni assegnamento mette A a 1; che A è insoddisdacibile se nessun assegnamento mette A a 1.

Definizione 1.5 (Conseguenza Logica)**.** Siano T un insieme di proposizioni e sia A una proposizione. Diciamo che A è *conseguenza logica* di T se ogni assegnamento che soddisfa tutti gli elementi di T soddisfa anche A. Scriviamo in tal caso $T \models A$ e diciamo che le premesse T implicano logicamente la conclusione A.

Se T è finito e consiste delle formule $A_1, \ldots, A_n$ scriviamo $A_1, \ldots, A_n \models A$ invece di $\{A_1, \ldots, A_n\} \models A$.

Se T è insoddisfacibile, e A è una proposizione qualunque, dalla definizione di $\models$ segue che $T \models A$. Infatti quest'ultima relazione significa: Per ogni assegnamento α, se α soddisfa T allora α soddisfa A. Questa implicazione è vera a vuoto, se nessun assegnamento soddisfa T. Ciò corrisponde all'idea che una teoria insoddisfacibile, ossia le cui ipotesi non possono mai verificarsi simultaneamente (qualunque cosa esse significhino), è una teoria del tutto inutile, che permette di concludere tutto e il contrario di tutto.

Se T è vuoto scriviamo $\models A$ invece di $\varnothing \models A$. In questo caso la definizione, letta correttamente, dice che A è soddisfatta da tutti gli assegnamenti, i.e., che per ogni assegnamento α, $\alpha(A) = 1$. Infatti per qualunque α è **vero a vuoto** che α soddisfa tutti gli elementi dell'insieme $\varnothing$ (per ogni B, se $B \in \emptyset$, allora $\alpha(B) = 1$ è vera a vuoto). Possiamo dire, in questo caso, che A è vera in virtù della sua forma o struttura logica, indipendentemente da altre ipotesi (chiameremo una A di questo tipo una *tautologia*).

Osservazione 1.6. Occorre osservare una importante differenza tra la nozione di implicazione $A \to B$ e quella di conseguenza logica $A \models B$. Date due formule arbitrarie A e B e un assegnamento α, è sempre vero che $\alpha(A \to B) = 1$ oppure $\alpha(B \to A) = 1$. Al contrario, non è sempre vero che $A \models B$ oppure $B \models A$. Per esempio, se p_1 e p_2 sono variabili proposizionali, non vale né $p_1 \models p_2$ né $p_2 \models p_1$ perché posso definire un assegnamento che verifica p_1 ma non p_2 e viceversa. D'altra parte, invece, per ogni assegnamento α, una tra $(p_1 \to p_2)$ e $(p_2 \to p_1)$ è necessariamente vera.

La nozione di conseguenza logica formalizza l'idea seguente: un argomento con ipotesi T e conseguenza A è valido se A è vera ogni volta che sono vere le ipotesi T.

La nozione di conseguenza logica di una teoria rispetta alcune proprietà del normale ragionamento matematico. Si osservi che le proprietà qui sotto valgono in particolare quando T è la teoria vuota.

Proposizione 1.7. *La relazione $\models$ di conseguenza logica gode delle seguenti proprietà elementari.*

(1) $T, A \models A$.
(2) $T, A \models (A \vee B)$
(3) $T, B \models (A \vee B)$
(4) $T, (A \wedge B) \models A$.
(5) *Se* $T, A \models B$ *e* $T, A \models C$ *allora* $T, A \models (B \wedge C)$.
(6) *Se* $T, A \models B$ *e* $T, B \models C$ *allora* $T, A \models C$.
(7) $T, A \models B$ *se e solo se* $T \models (A \to B)$.
(8) *Se* $T, A \models B$ *e* $T, \neg A \models B$ *allora* $T \models B$.
(9) *Se* $T, A \models C$ *e* $T, B \models C$ *allora* $T, A \vee B \models C$.

Dimostrazione. Dimostriamo il punto 6. Procediamo per assurdo. Assumiamo dunque che $T, A \models B$ e $T, B \models C$ ma che non valga $T, A \models C$. Scriviamo quest'ultimo fatto come $T, A \not\models C$. Per definizione significa che: Esiste un assegnamento α tale che $\alpha(T) = 1 = \alpha(A)$ ma $\alpha(C) = 0$. Sia α^* un tale assegnamento. Per l'ipotesi $T, A \models B$ abbiamo che: Per ogni assegnamento α, se $\alpha(T) = 1 = \alpha(A)$ allora $\alpha(B) = 1$. In particolare $\alpha^*(B) = 1$, dato che per ipotesi $\alpha^*(T) = 1 = \alpha^*(A)$. Per la seconda ipotesti $T, B \models C$ abbiamo che: Per ogni assegnamento α, se $\alpha(T) = 1 = \alpha(B)$ allora $\alpha(C) = 1$. Dato che $\alpha^*(T) = 1 = \alpha^*(B)$ segue che $\alpha^*(C) = 1$. Ma questo contraddice la scelta di α^*, per cui vale $\alpha^*(C) = 0$.

Consideriamo il punto 7. Assumiamo che $T, A \models B$ e dimostriamo $T \models (A \to B)$. Ragioniamo per assurdo, assumendo che $T \not\models (A \to B)$. Dunque esiste un assegnamento α tale che $\alpha(T) = 1$ e $\alpha(A \to B) = 0$. Dunque $\alpha(A) = 1$ e $\alpha(B) = 0$, per la semantica dell'implicazione. D'altra parte l'ipotesi dice che: Per ogni α, se $\alpha(T) = 1 = \alpha(A)$ allora $\alpha(B) = 1$. Ma questo è falsificato dall'assegnamento di cui abbiamo dimostrato l'esistenza. L'altra implicazione è lasciata al lettore.

Consideriamo il punto 8. Ragioniamo per assurdo assumendo che $T, A \models B$ e $T, \neg A \models B$ ma $T\neg \models A$. Per quest'ultimo fatto esiste un assegnamento α tale che $\alpha(T) = 1$ e $\alpha(B) = 0$. Per un tale α deve valere $\alpha(A) = 1$ o $\alpha(\neg A) = 1$. Nel primo caso abbiamo un α t.c. $\alpha(T) = 1 = \alpha(\neg A)$ e dunque dall'ipotesi $T, \neg A \models B$ segue che $\alpha(B) = 1$, contraddizione. Nel secondo caso abbiamo un α t..c. $\alpha(T) = 1 = \alpha(A)$ e dall'ipotesi $T, A \models B$ per questo α segue $\alpha(B) = 1$, in contraddizione con il fatto che $\alpha(B) = 0$ per scelta di α.

Il punto 9 è una forma generalizzata del punto 8 (Dimostrare per esercizio). □

La nozione di teoria soddisfacibile ammette due interessanti caratterizzazioni.

Proposizione 1.8. *Le seguenti proprietà sono equivalenti.*

(1) *T è soddisfacibile.*
(2) *Per ogni formula A, se $T \models A$ allora $T \not\models \neg A$.*
(3) *Esiste una formula A tale che $T \not\models A$.*

Inoltre, $T \models A$ se e solo se $T \cup \{\neg A\}$ è insoddisfacibile. Inoltre, se T è soddisfacibile, allora $T \cup \{A\}$ è soddisfacibile oppure $T \cup \{\neg A\}$ è soddisfacibile.

Dimostrazione. Sia T soddisfacibile. Per assurdo supponiamo che $T \models A$ e $T \models \neg A$. Allora esiste una valutazione v tale che $v(A) = v(\neg A) = 1$, il che è impossibile. Supposiamo (2) e supponiamo per assurdo che non valga (3). Allora T implica logicamente tutte le formule. In particolare per ogni A abbiamo $T \models A$ e $T \models \neg A$, contro il punto (2). Supponiamo (3) e dimostriamo (1). Per ogni formula A, $T \models A$ e $T \models \neg A$. Una valutazione modello di T è dunque impossibile.

Supponiamo ora che $T \models A$ e che $T \cup \{\neg A\}$ sia soddisfacibile. Preso un modello v di quest'ultima, dovremmo avere che $v(A) = 1$ e $v(\neg A) = 1$, il che è impossibile. Supponiamo ora che $T \cup \{\neg A\}$ sia insoddisfacibile. Non esistono valutazioni che soddisfano T e $\neg A$, dunque ogni valutazione che soddisfa T soddisfa anche A (si noti che possono non esistere valutazioni che soddisfano T ma il ragionamento ipotetico resta corretto).

Supponiamo ora che T sia soddisfacibile ma che per qualche A né $T \cup \{A\}$ né $T \cup \{\neg A\}$ siano soddisfacibili. Preso v che soddisfa T necessariamente deve valere $v(A) = 1$ oppure $v(A) = 0$. Nel primo caso v soddisfa $T \cup \{A\}$ e nel secondo caso v soddisfa $T \cup \{\neg A\}$. □

Per il seguente Teorema, il problema della conseguenza logica da un numero finito di premesse (validità di un argomento) è equivalente a quello della verità logica e della soddisfacibilità.

Teorema 1.9. *Siano $A_1, \ldots, A_n, A$ proposizioni. Allora i seguenti punti sono equivalenti.*

(1) $A_1, \ldots, A_n \models A$.
(2) $((A_1 \wedge \cdots \wedge A_n) \to A) \in$ Taut.
(3) $(A_1 \wedge \cdots \wedge A_n \wedge \neg A) \in$ UnSat.

Dimostrazione. Assumiamo il punto (1) e dimostriamo il (2). Supponiamo per assurdo che esista una valutazione v tale che $v((A_1 \wedge \cdots \wedge A_n) \to A) = 0$. Dunque $v(A_1 \wedge \cdots \wedge A_n)) = 1$ e $v(A) = 0$. Ma questo contraddice (1). Assumiamo il punto (2) e dimostriamo il (3). Per assurdo esista una valutazione v tale che $v(A_1 \wedge \cdots \wedge A_n \wedge \neg A) = 1$. Allora $v(A_1) = v(A_2) = \cdots = v(A_n) = v(\neg A) = 1$ e $v(A) = 0$, e dunque $v((A_1 \wedge \cdots \wedge A_n) \to A) = 0$, contro l'ipotesi che si tratti di una tautologia. Assumiamo il punto (3) e dimostriamo il punto (1). Per assurdo supponiamo che $A_1, \ldots, A_n \not\models A$. Allora esiste un assegnamento v tale che $v(A_1) = \cdots = v(A_n) = 1$ e $v(A) = 0$, e dunque $v(\neg A) = 1$ e così $v(A_1 \wedge \cdots \wedge A_n \wedge \neg A) = 1$, contro l'ipotesi. □

Vale una generalizzazione dell'equivalenza di punti (1) e (2) anche per la nozione di conseguenza logica da una teoria arbitraria.

Proposizione 1.10. *Se T è una teoria e $A_1, \ldots, A_n, B$ sono formule allora $T, A_1, \ldots, A_n \models B$ se e solo se $T \models (A_1 \wedge \cdots \wedge A_n) \to B$.*

Osservazione 1.11. Il metodo delle tavole di verità permette di calcolare i valori di verità di una funzione arbitrariamente complessa. Data una proposizione A qualunque, possiamo rispondere algoritmicamente alla domanda: $A \in$ TAUT? Basta costruire la tavola di verità di A e controllare se l'ultima colonna contiene solo il valore 1. La tavola di verità di una proposizione in cui appaiono n variabili proposizionali contiene 2^n righe. Per questo motivo il metodo delle tavole di verità è *computazionalmente inefficiente*. Lo stesso vale per la domanda: $A \in$ SAT? Anche in questo caso le tavole di verità danno una risposta, ma in modo inefficiente.

Non si conoscono però algoritmi efficienti (polinomiali) per rispondere a questa domanda. Trovare un tale algoritmo o dimostrare che un tale algoritmo non esiste equivale a risolvere il Problema del Millennio ($\mathbf{P} = \mathbf{NP}$)? (i.e., la classe dei problemi risolvibili in tempo polinomiale da un algoritmo deterministico coincide con la classe dei problemi risolvibili in tempo polinomiale da un algoritmo non-deterministico?). Per questo problema il *Clay Mathematical Institute* offre un premio di un milione di dollari.

In molti casi è possibile decidere se una certa proposizione è in TAUT o no, oppure se una certa conclusione è conseguenza logica di certe altre proposizioni senza costruire la tavola di verità, ma ragionando in modo rigoroso a un più alto livello. Nel seguito vediamo alcuni risultati che permettono di manipolare proposizioni in modo algebrico, preservando la relazione di equivalenza logica.

LOGICA MATEMATICA

A.A. 22/23, DISPENSA N. 4

SOMMARIO. Esempi di formalizzazioni in Logica Proposizionale.

1. Formalizzazioni in Logica Proposizionale

Esempio 1.1 (Puzzle logici)**.** I concetti introdotti finora si prestano bene alla risoluzione di puzzle logici di tipo enigmistico. Per esempio: vi sono tre persone A, B, C di cui una mente sempre mentre gli altri due dicono sempre il vero. A dice che B è il mentitore; B dice che C è il mentitore. Possiamo scoprire chi è il vero mentitore?

Introduciamo un linguaggio proposizionale con tre variabili: p_A (per: il mentitore è A), p_B (per: il mentitore è B) e p_C (per: il mentitore è C). Formalizziamo i vincoli dell'enigma.

Il vincolo che uno solo dei tre è mentitore si può spezzare in due componenti: **almeno uno è un mentitore**, e **non vi sono due mentitori**. Per il primo vincolo poniamo:

$$(p_A \vee p_B \vee p_C).$$

Come osservato in classe questa parte del vincolo può formalizzarsi anche con la seguente proposizione:

$$\neg(\neg p_A \wedge \neg p_B \wedge \neg p_C),$$

che esprime il concetto "non è possibile che né A né B né C siano mentitori". Come osservato in classe, questa proposizione è equivalente alla precedente.

per il secondo

$$\neg(p_A \wedge p_B) \wedge \neg(p_A \wedge p_C) \wedge \neg(p_B \wedge p_C).$$

L'affermazione di A si può formalizzare ponendo $\neg p_A \to p_B$, ossia: se A non mente allora B è il mentitore.

L'affermazione di B si può formalizzare ponendo $\neg p_B \to p_C$, ossia: se B non mente allora C è il mentitore.

Congiungendo i vari vincoli formalizzati otteniamo la seguente proposizione, che donotiamo con P:

$$(p_A \vee p_B \vee p_C) \wedge \neg(p_A \wedge p_B) \wedge \neg(p_A \wedge p_C) \wedge \neg(p_B \wedge p_C) \wedge (\neg p_A \to p_B) \wedge (\neg p_B \to p_C)$$

Scrivendo la tavola di verità di questa proposizione verifichiamo che è soddisfatta da un unico assegnamento α, quello per cui $\alpha(p_A) = 0$, $\alpha(p_B) = 1$ e $\alpha(p_C) = 0$. Dunque B è il mentitore.

In classe abbiamo discusso formalizzazioni alternative. In particolare è stata proposta la seguente proposizione come formalizzazione dei vincoli di contorno (esiste esattamente un mentitore):

$$Q = (p_A \wedge \neg p_B \wedge \neg p_C) \vee (\neg p_A \wedge p_B \wedge \neg p_C) \vee (\neg p_A \wedge \neg p_B \wedge p_C).$$

Domanda: la proposizione Q è equivalente a quella proposta in precedenza, ossia alla seguente proposizione P?

$$P = (p_A \vee p_B \vee p_C) \wedge \neg(p_A \wedge p_B) \wedge \neg(p_A \wedge p_C) \wedge \neg(p_B \wedge p_C)$$

Per il vincolo "esiste esattamente un mentitore" è stata proposta anche la seguente formalizzazione:

$$R = (p_A \to (\neg p_A \wedge \neg p_C)) \wedge (p_B \to (\neg p_A \wedge \neg p_C)) \wedge (p_C \to (\neg p_A \wedge \neg p_B)).$$

Domanda: Questa proposizione è equivalente alle precedenti?

Note preparate da Lorenzo Carlucci, lorenzo.carlucci@uniroma1.it.

Osservazione 1.2. Si osserva che l'esempio di sopra ammette una soluzione formale perché è ben posto. Si consideri, per esercizio, l'enigma modificato assumendo che B asserisca che A è il mentitore. Cosa ci dice l'analisi della tavola di verità della proposizione risultante? Consideriamo poi la modifica in cui, oltre alle ipotesi originali, assumiamo che C asserisca che A è il mentitore. Cosa ci dice l'analisi della tavola di verità della proposizione risultante?

Esempio 1.3 (Ordini)**.** Il concetto di soddisfacibilità permette di usare insiemi di formule proposizionali per catturare determinate strutture matematiche.

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

(1) $\neg p_{x,x}$ per ogni $x \in X$.
(2) $p_{x,y} \to \neg p_{y,x}$ per ogni $x, y \in X$.
(3) $(p_{x,y} \wedge p_{y,z}) \to p_{x,z}$ per ogni $x, y, z \in X$
(4) $p_{x,y} \vee p_{y,x}$ per ogni $x, y \in X$ con $x \neq y$.

Si noti che l'insieme appena descritto dipende dall'insieme X (per sottolinearlo possiamo scrivere T_X).

L'insieme $T = T_X$ *cattura* o *esprime* il concetto di ordine totale (stretto) su X nel senso seguente.

Supponiamo di avere un assegnamento α che soddisfa tutte le proposizioni di T.

Allora l'ordine *indotto* da tutte le variabili proposizionali vere sotto un tale assegnamento è un ordine totale stretto su X. Più precisamente, se α è un assegnamento, definiamo la relazione $\prec_\alpha$ su X come segue:

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

Si verifica facilmente che se $\alpha(T) = 1$ allora $\prec_\alpha$ è un ordine totale stretto su X.

D'altra parte, sia $\prec$ un ordine totale (stretto) su X. Consideriamo l'assegnamento $\alpha_\prec$ *indotto* da $\prec$ così definito:

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

Si verifica facilmente che $\alpha_\prec(T) = 1$.

Riassumendo possiamo dire che, per ogni insieme X, la teoria T_X è una buona formalizzazione del concetto di ordine totale stretto su X perché valgono i seguenti due punti:

(1) Per ogni assegnamento α che soddisfa T_X, l'ordine $\prec_\alpha$ indotto da α è un ordine totale stretto su X.
(2) Per ogni ordine totale stretto $\prec$ su X, l'assegnamento $\alpha_\prec$ indotto da $\prec$ sulle variabili $p_{x,y}$ soddisfa T.

I due punti precedenti implicano che: un assegnamento α soddisfa la teoria T_X se e soltanto se l'ordine indotto da α su X è un ordine totale.

Esempio 1.4 (Colorabilità di Grafi)**.** Consideriamo il problema: si può colorare la mappa in figura usando due colori (Rosso e Blu) rispettando il vincolo che due stati adiacenti hanno colori diversi?

Per iniziare, consideriamo il sottoproblema relativo alla sotto-mappa M contenente soltanto Italia, Austria e Ungheria. Dichiariamo il seguente linguaggio proposizionale: le variabili proposizionali sono $I_R, I_B, A_R, A_B, U_R, U_B$ e il loro significato intuitivo è I_R = l'Italia è rossa, I_B = l'Italia è blu, etc.

Per esprimere il vincolo che ogni nazione riceve almeno un colore scriviamo:

$$(1) \quad (I_R \vee I_B) \wedge (A_R \vee A_B) \wedge (U_R \vee U_B)$$

Per esprimere il vincolo che ogni nazione riceve al più un colore scriviamo:

$$(2) \quad (I_R \to \neg I_B) \wedge (A_R \to \neg A_B) \wedge (U_R \to \neg U_B).$$

Per esprimere il vincolo che nazioni confinanti hanno colori diversi, scriviamo:

$$(3) \quad (I_R \to \neg A_R) \wedge (I_B \to \neg A_B) \wedge (A_R \to \neg U_R) \wedge (A_B \to \neg U_B).$$

Osserviamo che questa proposizione dipende dall'istanza del problema in questione ossia dai confini presenti nella mappa.

Ovviamente l'insieme di proposizioni (teoria) appena descritto *dipende* dalla mappa M. Intuitivamente questa teoria formalizza adeguatamente il problema della 2-colorazione per la mappa M.

Tecnicamente questo significa che l'insieme delle proposizioni scritte sopra è in SAT se e soltanto se la mappa di Italia, Austria e Ungheria è 2-colorabile (ossia colorabile in 2 colori rispettando il vincolo).

Supponiamo infatti che sia 2-colorabile. Allora esiste un assegnamento di colori Rosso, Blu a Italia, Austria e Ungheria che rispetta il vincolo. Formalmente si tratta di una funzione:

$$f : \{\text{Italia},\ \text{Austria},\ \text{Ungheria}\} \to \{\text{Rosso},\ \text{Blu}\}.$$

Se questa colorazione è Italia Rossa, Austria Blu, Ungheria Rossa, definiamo l'assegnamento proposizionale $\alpha(I_R) = 1$, $\alpha(I_B) = 0$, $\alpha(A_R) = 0$, $\alpha(A_B) = 1$, $\alpha(U_R) = 1$, $\alpha(U_B) = 0$. In generale una colorazione f induce un assegnamento α_f alle variabili del nostro linguaggio. Si verifica facilmente che questo assegnamento soddisfa tutte le proposizioni usate per formalizzare il problema.

Viceversa, dato un assegnamento α che mette a 1 tutte le proposizioni usate per la formalizzazione del problema, posso estrarre una colorazione: se $\alpha(I_R) = 1$ allora coloro l'Italia di Rosso, se $\alpha(A_B) = 1$ coloro l'Austria di Blu, e così via. Dato che l'assegnamento soddisfa, per es., la formula $I_R \to \neg I_B$, non assegnerò due colori distinti all'Italia. Si verifica facilmente che il fatto che α soddisfa tutte le formule in questione implica che la colorazione ottenuta soddisfai vincoli per essere una soluzione corretta al problema della 2-colorazione.

Abbiamo quindi i due punti seguenti:

(1) Ogni colorazione f in 2 colori che soddisfa il vincolo induce un assegnamento che soddisfa la teoria,
(2) Ogni assegnamento che soddisfa la teoria induce una colorazione in 2 colori che soddisfa il vincolo.

Dai due punti seguenti segue che la mappa M è 2-colorabile se e soltanto se la teoria T_M è soddisfacibile.

Osservazione 1.5. La formalizzazione sopra introdotta non è la più economica possibile. Osserviamo infatti che il vincolo espresso al punto (2) è implicato dai vincoli (3) e (1) ed è pertanto ridondante (per quanto non nocivo). In termini formali verifichiamo che (2) è conseguenza logica di (1) e (2), ossia

$$(I_R \vee I_B) \wedge (A_R \vee A_B) \wedge (U_R \vee U_B), (I_R \to \neg A_R) \wedge (I_B \to \neg A_B) \wedge (A_R \to \neg U_R) \wedge (A_B \to \neg U_B)$$
$$\models (I_R \to \neg I_B) \wedge (A_R \to \neg A_B) \wedge (U_R \to \neg U_B).$$

(Esercizio)

Consideriamo ancora il problema della 2-colorazione per Slovenia, Austria e Ungheria. Analogamente a quanto fatto sopra usiamo un vocabolario proposizionale composto dalle variabili $A_R, A_B, S_R, S_B, U_R, U_B$ e formalizziamo con le seguenti proposizioni:

$$(S_R \vee S_B) \wedge (A_R \vee A_B) \wedge (U_R \vee U_B).$$
$$(S_R \to \neg S_B) \wedge (A_R \to \neg A_B) \wedge (U_R \to \neg U_B).$$
$$(S_R \to \neg A_R) \wedge (S_B \to \neg A_B) \wedge (S_R \to \neg U_R) \wedge (S_B \to \neg U_B) \wedge (A_R \to \neg U_R) \wedge (A_B \to \neg U_B)$$

Si verifica che l'insieme delle proposizioni qui sopra (o equivalentemente la loro congiunzione) è insoddisfacibile. Invece di usare una tavola di verità si può ragionare, per es. così: Supponiamo che α soddisfi tutte le proposizioni di sopra. Supponiamo che $\alpha(S_R) = 1$. Allora $\alpha(\neg A_R) = 1$ e dunque $\alpha(A_B) = 1$. Ma allora $\alpha(\neg U_B) = 1$ e dunque $v(U_R) = 1$. Ma allora $\alpha(S_R \to \neg U_R) = 0$. Contraddizione.

La formalizzazione è adeguata perché, come nell'esempio precedente, l'insieme di formule è soddisfacibile se e solo se esiste una soluzione al problema della 2-colorazione della mappa.

Generalizzando gli esempi di sopra vediamo come formalizzare in logica proposizionale un problema di colorabilità di grafi generale.

Fissato un grafo $G = (V, E)$, consideriamo il seguente linguaggio proposizionale. Per ogni $v \in V$ e per ogni $i \in [1, k]$ abbiamo una variabile proposizionale $P_{v,i}$. Il significato intuitivo di questa variabile è che v ha il colore i. Consideriamo il seguente insieme T di formule:

(1) $P_{v,1} \vee P_{v,2} \vee \cdots \vee P_{v,k}$, per ogni $v \in V$.
(2) $\neg(P_{v,i} \wedge P_{v,j})$ per ogni $v \in V$, $i \neq j$ in $[1, k]$.
(3) $\neg(P_{v,i} \wedge P_{w,i})$ per ogni $\{v, w\} \in E$, $i \in [1, k]$.

Si osserva che se il grado G è finito allora T è un insieme finito di formule e si può considerare la singola formula ottenuta congiungendo tutte le formule in T. Se G è infinito allora T è un insieme infinito di formule.

Sia α un assegnamento booleano alle variabili del linguaggio. Se α soddisfa T, possiamo definire una colorazione di G in k colori come segue.

$$c(v) = i \quad \text{se e solo se } \alpha(P_{v_i}) = 1.$$

Si dimostra facilmente che c è una k-colorazione di G. Viceversa, una k-colorazione di G induce un assegnamento che soddisfa T.

Esempio 1.6 (Pigeonhole Principle)**.** Questo esempio illustra come formalizzare in logica proposizionale principi matematici in cui appaiono quantificatori (per ogni, esiste) ma solo su un numero finito di oggetti, dando luogo a formule soddisfatte da tutti gli assegnamenti.

Consideriamo il Principio dei Cassetti (o dei Piccioni, *Pigeonhole Principle*).

> **Principio dei Cassetti**. Per ogni $n \in \mathbf{N}$, $n \geq 1$, se ho messo $n + 1$ oggetti in n cassetti allora almeno un cassetto contiene più di un oggetto.

Indichiamo questo principio, per ogni $n \geq 1$ fissato, con $PHP(n + 1, n)$. Ovviamente un analogo principio vale se usiamo un qualunque $m \geq n + 1$ al posto di $n + 1$. In termini più matematici, $PHP(n + 1, n)$ si può esprimere come segue.

> **PHP**$(\mathbf{n + 1, n})$. Se f è una funzione suriettiva con dominio $\{1, \ldots, n + 1\}$ e codominio $\{1, \ldots, n\}$, allora esiste un elemento del codominio che ha almeno due preimmagini via f.

In altre parole, non esiste una funzione iniettiva con dominio $\{1, \ldots, n + 1\}$ e codominio $\{1, \ldots, n\}$.

Si osserva facilmente che il principio vale anche se f è una relazione $\subseteq [1, 4] \times [1, 3]$ ovunque definita, nel senso che per ogni $x \in [1, 4]$ esiste (almeno) un $y \in [1, 3]$ tale che $(x, y) \in f$.

Per ogni scelta di n, facciamo vedere come formalizzare $PHP(n + 1, n)$ nel linguaggio proposizionale.

Fissiamo per semplicità $n = 3$. Vogliamo formalizzare $PHP(4, 3)$, che dice che se f è una realzione ovunque definita sul dominio $\{1, 2, 3, 4\}$ e codominio $\{1, 2, 3\}$ allora un elemento del dominio ha almeno due preimmagini secondo f.

Spezziamo questo enunciato in due parti.

(1) f è una relazione con dominio $\{1, 2, 3, 4\}$ e codominio $\{1, 2, 3\}$ (ovunque definita).
(2) f non è iniettiva.

Dobbiamo formalizzare: **Se** f è una funzione con dominio $\{1, 2, 3, 4\}$ e codominio $\{1, 2, 3\}$ **allora** f non è iniettiva. Ossia **Se** (1) allora (2).

Un linguaggio adeguato per formalizzare $PHP(4, 3)$ è il linguaggio che ha come variabili proposizionali i simboli $p_{i,j}$, dove i varia in $\{1, 2, 3, 4\}$ e j varia in $\{1, 2, 3\}$. Dunque le variabili del linguaggio sono 12 in tutto. (N.B.B. I simboli $p_{i,j}$ non fanno parte del linguaggio!! Sono solo un modo comodo per quantificare su $\{1, 2, 3, 4\}$ e $\{1, 2, 3\}$. Le vere variabili sono simboli del tipo $p_{1,1}$, $p_{4,2}$, $p_{4,3}$ etc.). Il significato *intuitivo* delle variabili scelte è il seguente.

$$p_{i,j} \text{ sta per } f(i) = j.$$

Cominciamo formalizzando (1). Vogliamo esprimere il fatto che ogni elemento del dominio $\{1, 2, 3, 4\}$ viene associato ad almeno un elemento del codominio, ossia che f è una relazione in $\{1, 2, 3, 4\} \times \{1, 2, 3\}$ ovunque definita. In termini comuni la proprietà in questione si esprime così: Per ogni $i \in \{1, 2, 3, 4\}$ esiste un $j \in \{1, 2, 3\}$ tale che $f : i \mapsto j$. Dato che abbiamo a che fare con insiemi finiti, possiamo svolgere i quantificatori universali come congiunzioni e gli esistenziali come disgiunzioni. Per ogni $i \in \{1, 2, 3, 4\}$ dà luogo a quattro proposizioni:

- $(i = 1)$: Esiste un $j \in \{1, 2, 3\}$ tale che $f : 1 \mapsto j$.
- $(i = 2)$: Esiste un $j \in \{1, 2, 3\}$ tale che $f : 2 \mapsto j$.
- $(i = 3)$: Esiste un $j \in \{1, 2, 3\}$ tale che $f : 3 \mapsto j$.
- $(i = 4)$: Esiste un $j \in \{1, 2, 3\}$ tale che $f : 4 \mapsto j$.

Consideriamo la prima e osserviamo che l'esistenziale si può rappresentare con una disgiunzione sui possibili valori di j:

$$(p_{1,1} \vee p_{1,2} \vee p_{1,3}).$$

Analogamente per le altre tre proposizioni:

$$(p_{2,1} \vee p_{2,2} \vee p_{2,3}).$$
$$(p_{3,1} \vee p_{3,2} \vee p_{3,3}).$$
$$(p_{4,1} \vee p_{4,2} \vee p_{4,3}).$$

Il quantificatore universale in (1) (per ogni $i \in \{1, 2, 3, 4\}$ viene infine espresso congiungendo le quattro proposizioni ottenute sopra:

$$A = (p_{1,1} \vee p_{1,2} \vee p_{1,3}) \wedge (p_{2,1} \vee p_{2,2} \vee p_{2,3}) \wedge (p_{3,1} \vee p_{3,2} \vee p_{3,3}) \wedge (p_{4,1} \vee p_{4,2} \vee p_{4,3}).$$

Abbreviamo per comodità con $\bigwedge_{i=1}^{n} A_i$ (risp. $\bigvee_{i=1}^{n} A_i$) la proposizione $A_1 \wedge \cdots \wedge A_n$ (risp. $A_1 \vee \cdots \vee A_n$). Con questa notazione possiamo scrivere la congiunzione di sopra come

$$\bigwedge_{i=1}^{4} (p_{i,1} \vee p_{i,2} \vee p_{i,3})$$

In modo ancora pù sintetico possiamo scriverla come segue

$$\bigwedge_{i=1}^{4} \bigvee_{j=1}^{3} p_{i,j}.$$

Ora formalizziamo (2), partendo dalla formalizzazione di f è iniettiva e applicando una negazione. f è iniettiva se e solo se non esiste un elemento del codominio con due pre-immagini distinte secondo f. In altre parole, f è iniettiva se e solo se per ogni elemento j del codominio, per ogni scelta di due pre-immagini distinte i, i' nel dominio, non è vero che $f(i) = j$ e $f(i') = j$. Dobbiamo quindi formalizzare l'enunciato seguente.

$$\forall j \in \{1, 2, 3\} \forall i \neq i' \in \{1, 2, 3, 4\} (f(i) \neq j \vee f(i') \neq j).$$

Procediamo come sopra. Consideriamo uno per uno i valori di j.

Per $j = 1$, dobbiamo formalizzare

$$\forall i \neq i' \in \{1, 2, 3, 4\} (f(i) \neq 1 \vee f(i') \neq 1).$$

Per ognuna delle $\binom{4}{2}$ scelte di due elementi distinti $i, i' \in \{1, 2, 3, 4\}$ dobbiamo formalizzare $(f(i) \neq 1 \vee f(i') \neq 1)$. Quest'ultimo enunciato si formalizza ovviamente con $\neg(p_{i,1} \wedge p_{i',1})$ (o equivalentemente con $(\neg p_{i,1} \vee \neg p_{i',1})$). Dato che la quantificazione su i, i' è universale, otteniamo la seguente congiunzione.

$$\neg(p_{1,1} \wedge p_{2,1}) \wedge \neg(p_{1,1} \wedge p_{3,1}) \wedge \neg(p_{1,1} \wedge p_{4,1}) \wedge \neg(p_{2,1} \wedge p_{3,1}) \wedge \neg(p_{2,1} \wedge p_{4,1}) \wedge \neg(p_{3,1} \wedge p_{4,1}).$$

Analogamente, per $j = 2$ otteniamo

$$\neg(p_{1,2} \wedge p_{2,2}) \wedge \neg(p_{1,2} \wedge p_{3,2}) \wedge \neg(p_{1,2} \wedge p_{4,2}) \wedge \neg(p_{2,2} \wedge p_{3,2}) \wedge \neg(p_{2,2} \wedge p_{4,2}) \wedge \neg(p_{3,2} \wedge p_{4,2}).$$

Analogamente, er $j = 3$ otteniamo

$$\neg(p_{1,3} \wedge p_{2,3}) \wedge \neg(p_{1,3} \wedge p_{3,3}) \wedge \neg(p_{1,3} \wedge p_{4,3}) \wedge \neg(p_{2,3} \wedge p_{3,3}) \wedge \neg(p_{2,3} \wedge p_{4,3}) \wedge \neg(p_{3,3} \wedge p_{4,3}).$$

Infine, per esprimere la quantificazione universale $\forall j \in \{1, 2, 3\} \ldots$ basta prendere la congiunzione delle tre proposizioni ottenute per i singoli valori di j.

$$\begin{aligned} B = {} & \neg(p_{1,1} \wedge p_{2,1}) \wedge \neg(p_{1,1} \wedge p_{3,1}) \wedge \neg(p_{1,1} \wedge p_{4,1}) \wedge \neg(p_{2,1} \wedge p_{3,1}) \wedge \neg(p_{2,1} \wedge p_{4,1}) \wedge \neg(p_{3,1} \wedge p_{4,1}) \wedge \\ & \neg(p_{1,2} \wedge p_{2,2}) \wedge \neg(p_{1,2} \wedge p_{3,2}) \wedge \neg(p_{1,2} \wedge p_{4,2}) \wedge \neg(p_{2,2} \wedge p_{3,2}) \wedge \neg(p_{2,2} \wedge p_{4,2}) \wedge \neg(p_{3,2} \wedge p_{4,2}) \wedge \\ & \neg(p_{1,3} \wedge p_{2,3}) \wedge \neg(p_{1,3} \wedge p_{3,3}) \wedge \neg(p_{1,3} \wedge p_{4,3}) \wedge \neg(p_{2,3} \wedge p_{3,3}) \wedge \neg(p_{2,3} \wedge p_{4,3}) \wedge \neg(p_{3,3} \wedge p_{4,3}). \end{aligned}$$

Questa proposizione esprime l'iniettività; dunque $\neg B$ esprime la non-iniettività.

Per concludere, possiamo formalizzare $PHP(4,3)$ formalizzando: Se (1) allora (2), ossia $(A \to \neg B)$.
In forma sintetica la formula proposta è la seguente:

$$\bigwedge_{i=1}^{4} \bigvee_{k=1}^{2} p_{i,k} \to \neg \left(\bigwedge_{k=1}^{3} \bigwedge_{i \neq j, i,j \in [1,4]} (\neg p_{i,k} \vee \neg p_{j,kj}) \right).$$

L'antecedente esprime il fatto che ogni elemento in $[1,4]$ viene associato ad almeno un elemento in $[1,3]$. Il conseguente esprime la negazione dell'iniettività dell'associazione in questione.

La formalizzazione qui sopra non è l'unica possibile. Si possono ottenere formalizzazione altrettanto valide ma sintatticamente differenti partendo da descrizioni verbalmente differenti del fatto che f è iniettiva:

- Non esiste $k \in [1,3]$ tale che esistono $i \neq j$ in $[1,4]$ tali che $f(i) = k = f(j)$,
- Per ogni $k \in [1,3]$ per ogni $j \in [1,4]$, se $f(j) = k$ allora per ogni $i \neq j$, $i \in [1,4]$ vale $f(i) \neq k$.

Si invita il lettore a scrivere le versioni formali delle due frasi precedenti.

Tanta fatica per formalizzare un singolo caso del Principio dei Cassetti? Osserviamo che la formalizzazione svolta sopra è *uniforme* nel senso che se volessimo formalizzare $PHP(101,100)$ o $PHP(2^9, 2^9-1)$ potremmo usare lo stesso procedimento. Avremmo proposizioni più lunghe ma di stessa struttura.

Le formule risultanti saranno vere sotto ogni assegnamento, come si può facilmente stabilire invocando proprio il Principio dei Cassetti. Chiamiamo una tale formula una tautologia. Analogamente, se consideriamo la formalizzazione della negazione del PHP, ossia f associa ogni elemento di $[n+1]$ a un elemento di $[n]$ e f è iniettiva, otteniamo una formula sempre falsa. Chiamiamo una tale formula una contraddizione o una formula insoddisfacibile.

Possiamo esprimere una forma debole della **negazione** del PHP per m piccioni e n cassetti con il seguente insieme di formule nel linguaggio composto da variabili $p_{i,j}$:

(1) $p_{i,1} \vee \cdots \vee p_{i,n}$ per ogni $i \in [1,m]$, (in forma sintetica $\bigwedge_{i=1}^{4} \bigvee_{j=1}^{n} p_{i,j}$)
(2) $\neg p_{i,k} \vee \neg p_{j,k}$ per ogni $i,j \in [1,m]$ con $i \neq j$ e per ogni $k \in [1,n]$.

In forma sintetica abbiamo:

$$\bigwedge_{i=1}^{4} \bigvee_{j=1}^{n} p_{i,j} \wedge \bigwedge_{i \neq j, i,j \in [1,m]} \bigwedge_{k=1}^{n} (\neg p_{i,k} \vee \neg p_{j,k}).$$

L'insieme di formule qui sopra (o, equivalentemente la singola formula ottenuta congiungendo tutte le formule qui sopra) esprime la negazione del Principio dei Cassetti per i valori n e m e se $n < m$ è falsa sotto ogni assegnamento (insoddisfacibile).

Esercizio 1.7 (Sudoku)**.** Consideriamo il gioco del Sudoku su una tabella 9×9.

1	4	**5**	**3**	2	7	6	9	8
8	3	9	6	5	4	1	**2**	7
6	**7**	2	9	**1**	8	**5**	4	3
4	9	6	1	8	**5**	**3**	7	2
2	**1**	8	4	**7**	3	9	5	**6**
7	5	**3**	**2**	9	6	4	**8**	1
3	**6**	7	**5**	4	2	8	1	**9**
9	8	**4**	7	6	1	2	**3**	5
5	2	1	8	3	**9**	**7**	6	4

Vogliamo definire un insieme di proposizioni tali che gli assegnamenti che le soddisfano corrispondano esattamente alle soluzioni del gioco.

LOGICA MATEMATICA

A.A. 23/24, DISPENSA N. 5

SOMMARIO. Teorema di Compattezza proposizionale. Formulazione e dimostrazione per linguaggi numerabili.

1. Teorema di Compattezza

Si osserva facilmente che se $A_1, \dots, A_n \models A$ (una proposizione A segue logicamente dalle proposizioni $A_1, \dots, A_n$) allora $T \models A$ per ogni teoria T contenente $A_1, \dots, A_n$. Diciamo che la nozione di conseguenza logica è *monotona*, ossia che se $T \models A$ e $T' \supseteq T$ allora anche $T' \models A$; come segue facilmente dalle definizioni.

Consideriamo l'implicazione inversa. Assumiamo cioè che $T \models A$. Per una teoria T infinita, questo significa asserire che A è vera in tutti i modelli in cui tutte le infinite proposizioni di T sono vere. Nulla ci suggerisce a priori che in questo caso *esista* un sottinsieme finito $\{A_1, \dots, A_n\}$ che implichi logicamente A. Non sembra implausibile che *per ogni* sottinsieme finito $\{A_1, \dots, A_n\}$ di T valga $A_1, \dots, A_n \not\models A$. Questo significa soltanto che esiste un α che verifica $A_1, \dots, A_n$ ma falsifica A. Affinché questo non contraddica la nostra ipotesi che $T \models A$ basta che un tale α falsifichi *qualche altra* proposizione in T. Nulla sembra vietare che questo possa accadere.

Sorprendentemente vale il seguente teorema:

Teorema 1.1 (Teorema di Compattezza, versione 1)**.** *Se $T \models A$ allora esiste un sottinsieme finito T_0 di T tale che $T_0 \models A$.*

Consideriamo ora il caso di una teoria infinita T soddisfacibile. Questo significa che *esiste* un assegnamento che soddisfa *simultaneamente* tutte le proposizioni in T. Ovviamente se una tale assegnamento esiste allora è anche vero che *ogni sottinsieme finito* di T è soddisfacibile (dal medesimo assegnamento che soddisfa l'intera T). Chiamiamo *finitamente soddisfacibile* una teoria che possiede la proprietà appena formulata.

Definizione 1.2 (Teoria finitamente soddisfacibile)**.** Una teoria è *finitamente soddisfacibile* se ogni suo sottinsieme finito è soddisfacibile.

Consideriamo ora la domanda inversa: se T è finitamente soddisfacibile, è vero che T è soddisfacibile? L'ipotesi ci assicura che *per ogni scelta* di un sottinsieme finito di proposizioni in T *esiste* un assegnamento che lo soddisfa. Ma ovviamente questo assegnamento può essere diverso per ogni scelta del sottinsieme finito. Non abbiamo garanzie a priori circa la mutua compatibilità di questi assegnamenti. La conclusione richiede di dimostrare l'esistenza di un unico assegnamento che soddisfa simultaneamente tutte le infinite proposizioni in T. Dimostreremo il seguente Teorema.

Teorema 1.3 (Teorema di Compattezza, versione 2)**.** *Se ogni sottinsieme finito di T è soddisfacibile allora T è soddisfacibile.*

Cominciamo osservando che si tratta di una riformulazione del Teorema precedente. Si dimostri per esercizio la seguente proposizione.

Proposizione 1.4. *I due punti seguenti sono equivalenti:*

1. *$T \models A$ se e solo se esiste un sottinsieme finito T_0 di T tale che $T_0 \models A$.*
2. *T è soddisfacibile se e solo se T è finitamente soddifacibile.*

Dimostrazione. Esercizio. □

Note preparate da Lorenzo Carlucci, lorenzo.carlucci@uniroma1.it.

Si osserva facilmente che la nozione di soddisfacibilità gode della seguente proprietà di estendibilità: posso aggiungere una qualunque formula o la sua negazione a una teoria soddisfacibile preservandone la soddisfacibilità.

Proposizione 1.5 (Estendibilità)**.** *Se T è soddisfacibile, allora $T \cup \{A\}$ è soddisfacibile oppure $T \cup \{\neg A\}$ è soddisfacibile.*

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

La nozione di teoria finitamente soddisfacibile gode di una proprietà di estensione analoga a quella sopra osservata per la soddisfacibilità. Questa proprietà risulterà cruciale nella dimostrazione del Teorema di Compattezza.

Proposizione 1.6. *Sia T finitamente soddisfacibile. Per ogni formula A, $T \cup \{A\}$ è finitamente soddisfacibile o $T \cup \{\neg A\}$ è finitamente soddisfacibile.*

Dimostrazione. Ragioniamo per assurdo. Se né $T \cup \{A\}$ né $T \cup \{\neg A\}$ sono finitamente soddisfacibili, allora esistono sottinsiemi finiti $\{B_1, \dots, B_n\}$ di $T \cup \{A\}$ e $\{C_1, \dots, C_m\}$ di $T \cup \{\neg A\}$ insoddisfacibili. Dato che T è finitamente soddisfacibile per ipotesi, deve valere $A \in \{B_1, \dots, B_n\}$ e $\neg A \in \{C_1, \dots, C_m\}$. Dunque l'insieme $(\{B_1, \dots, B_n\} \setminus \{A\}) \cup (\{C_1, \dots, C_m\} \setminus \{\neg A\}) \subseteq T$ e quindi è finitamente soddisfacibile. Sia α un assegnamento che lo soddisfa. Se $\alpha(A) = 1$ allora α soddisfa anche $\{B_1, \dots, B_n\}$. Se $\alpha(A) = 0$ allora α soddisfa anche $\{C_1, \dots, C_m\}$. In entrambi i casi abbiamo una contraddizione. □

Dimostriamo ora il Teorema di Compattezza nella formulazione in termini di soddisfacibilità. Trattiamo prima il caso di linguaggi numerabili, poi il caso generale.

2. Teorema di Compattezza per Linguaggi numerabili

Teorema 2.1 (Teorema di Compattezza per linguaggi numerabili)**.** *Sia T in un linguaggio numerabile. Se T è finitamente soddisfacibile allora T è soddisfacibile.*

Dimostrazione. Assumiamo che il linguaggio di T sia $\{p_1, p_2, p_3, \dots\}$ e assumiamo di aver fissato una enumerazione delle variabili proposizionali.

Definiamo una successione di teoria come segue. $T_0 = T$, e poniamo T_{n+1} uguale a $T_n \cup \{p_{n+1}\}$ se questa teoria è finitamente soddisfacibile e uguale a $T_n \cup \{\neg p_{n+1}\}$ altrimenti.

Osserviamo che la definizione è ben posta in quanto se una teoria T è finitamente soddisfacibile, e A è una arbitraria proposizione, almeno una tra $T \cup \{A\}$ e $T \cup \{\neg A\}$ è finitamente soddisfacibile.

Ovviamente $T_0 \subseteq T_1 \subseteq T_2 \subseteq \dots$.

Definiamo $T^* = \bigcup_{n \in \mathbf{N}} T_n$.

Si osserva facilmente che per ogni $n \in \mathbf{N}$, la teoria T_n è finitamente soddisfacibile.

Da questo segue che anche T^* è finitamente soddisfacibile. Sia S un sottinsieme finito di T^*, e supponiamo che $S = \{A_1, \dots, A_k\}$. Dato che $S \subseteq \bigcup_{n \in \mathbf{N}}$, per ogni $i \in [1, k]$ esiste un $n_i \in \mathbf{N}$ tale che $A_i \in T_{n_i}$. Sia $n*$ il massimo tra gli n_i. Allora $S \subseteq T_{n^*}$.

Osserviamo inoltre che, per costruzione, per ogni variabile p_n, vale che $p_n \in T^*$ o $\neg p_n \in T^*$ ma non entrambe.

Allora il seguente assegnamento risulta ben definito:

$$\alpha(p_n) = 1 \text{ se } p_n \in T^*,$$
$$\alpha(p_n) = 0 \text{ se } \neg p_n \in T^*.$$

Vogliamo ora dimostrare che questo assegnamento α soddisfa la teoria T da cui siamo partiti.

Sia A una proposizione in T. Siano $p_{i_1}, \dots, p_{i_t}$ tutte le variabili proposizionali che compaiono in A. Per $j \in \{i_1, \dots, i_t\}$ definiamo $\hat{p_j}$ come p_j se $p_j \in T^*$ e come $\neg p_j$ se $\neg p_j \in T^*$. Per le proprietà di T^* si dà una e una sola di queste possibilità. Abbiamo che $A \in T \subseteq T^*$ e $\{\hat{p_{i_1}}, \dots, \hat{p_{i_t}}\} \subseteq T^*$ per definizione. Dunque $S = A \cup \{\hat{p_{i_1}}, \dots, \hat{p_{i_t}}\}$ è un sottinsieme finito di T^*. Dato che T^* è finitamente soddisfacibile, esiste un assegnamento β che soddisfa S. Un tale β soddisfa in particolare $\{\hat{p_{i_1}}, \dots, \hat{p_{i_t}}\}$ e questo implica necessariamente che $\beta(p_j) = 1$ se $p_j \in T^*$ e $\beta(p_j) = 0$ se $\neg p_j \in T^*$, per definizione dei $\hat{p_j}$, per $j \in \{i_1, \dots, i_t\}$.

D'altra parte anche α soddisfa $\{\hat{p_{i_1}}, \ldots, \hat{p_{i_t}}\}$. Dunque α e β concordano su tutte le variabili proposizionali che compaiono in A. Da questo segue che $\alpha(A) = \beta(A)$. Ma $\beta(A) = 1$ per scelta di β, dunque $\alpha(A) = 1$.

La dimostrazione è così conclusa.

□

LOGICA MATEMATICA

A.A. 23/24, DISPENSA N. 6

SOMMARIO. Teorema di Compattezza proposizionale.

1. Teorema di Compattezza per Linguaggi arbitrari

Nella dimostrazione del Teorema per linguaggi numerabili abbiamo sfruttato fin dall'inizio la numerabilità del linguaggio, fissando una enumerazione $p_1, p_2, p_3, \dots$ delle variabili proposizionali. Questa possibilità ci è preclusa se il linguaggio di partenza non è numerabile. Abbiamo poi costruito una successione numerabile di teorie $T_0, T_1, T_2, \dots$ partendo da T (che assumiamo finitamente soddisfacibile) e aggiungendo a ogni passo una variabile o la sua negazione, a seconda che l'estensione con l'una o con l'altra risulti finitamente soddisdacibile. Le teorie così definite sono tutte finitamente soddisfacibili e sono ordinate in modo crescente per inclusione, ossia

$$T = T_0 \subseteq T_1 \subseteq T_2 \subseteq T_3 \subseteq \dots$$

Abbiamo infine considerato l'unione di tutte queste teorie e dimostrato che questa unione è essa stessa finitamente soddisfacibile e induce un assegnamento che soddisfa T. Il fatto che l'unione induca un assegnamento è ottenuto per costruzione: l'unione in questione prende posizione/ha una opinione su ciascuna variabile proposizionale (perché include o la variabile stessa o la sua negazione), e questo è sufficiente ad avere un'opnione (= assegnare un valore di verità) a una qualunque formula del linguaggio. Inoltre, l'assegnamento indotto dall'unione delle T_n è coerente con T.

Nel caso generale vogliamo ottenere una teoria che abbia proprietà simili a quelle che aveva l'unione $\bigcup_n T_n$ nel caso numerabile. Per questo utilizziamo il Lemma di Zorn, che ci garantisce l'esistenza di un oggetto massimale in qualunque ordine parziale che soddisfa una certa proprietà.

Teorema 1.1 (Lemma di Zorn)**.** *Sia X un insieme e $\leq \subseteq X^2$ una relazione di ordine parziale (i.e., riflessiva, antisimmetrica e transitiva) su X. Se per ogni catena C in X (i.e., per ogni sottinsieme di X i cui elementi sono due a due confrontabili via $\leq$) esiste un maggiorante in X (ossia un elemento $x \in X$ tale che per ogni $y \in C$ vale $y \leq x$), allora esiste un elemento $m \in X$ massimale (ossia tale che per ogni $y \in X$, se $m \leq y$ allora $y = m$).*

Il Lemma di Zorn è una forma dell'Assioma della Scelta. Trova applicazione in molte aree della Matematica e in dimotrazioni di risultati importanti (e.g. ogni campo ha una chiusura algebrica, ogni spazio vettoriale non banale ha una base, il Teorema di Tychonoff in Topologia, etc.).

Per noi è sufficiente considerare gli ordini parziali in cui X è una famiglia di sottinsiemi di un certo insieme A e la relazione d'ordine $\leq$ è l'inclusione insiemistica. In questo caso possiamo riformulare il Lemma come segue.

Teorema 1.2 (Lemma di Zorn per famiglie di insiemi)**.** *Sia A un insieme e $P(A)$ il suo insieme potenza. Sia $F \subseteq P(A)$ una famiglia di sottinsiemi di A. Se per ogni catena C in F (i.e., per ogni famiglia di sottinsiemi di A appartenenti a F i cui elementi sono due a due confrontabili via $\subseteq$) esiste un maggiorante in F (ossia un sottinsieme S di A in F tale che per ogni $S' \in C$ vale $S' \leq S$), allora esiste un sottinsieme M di A in F (ossia tale che per ogni $S \in F$, se $M \leq S$ allora $S = M$).*

Si osserva facilmente che se 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.

Note preparate da Lorenzo Carlucci, lorenzo.carlucci@uniroma1.it.

Esempio 1.3. Sia A un insieme finito, e.g. $A = \{1, 2, \ldots, 100\}$. L'insieme potenza $P(A)$ è ovviamente parzialmente ordinato per inclusione insiemistica. Risulta ovvio che $P(A)$ contiene un elemento massimale, ovvero A. Altrettanto ovviamente ogni catena di elementi in $P(A)$ ha un maggiorante in $P(A)$: se la catena è $S_1 \subseteq S_2 \subseteq \cdots \subseteq S_t$, allora $\bigcup_{i=1}^{t} S_i$ è in $P(A)$ ed è un maggiorante della catena: per ogni $i \in [1, t]$ vale $S_i \subseteq \bigcup_i S_i$.

Esempio 1.4. Sia $A = \mathbf{N}$ e consideriano la famiglia F dei sottinsiemi finiti di $\mathbf{N}$. Ovviamente l'inclusione è un ordine parziale su F. Si vede facilmente che F non ha un elemento massimale: ogni sottinsieme finito di $\mathbf{N}$ può essere esteso propriamente a un altro sottinsieme finito di $\mathbf{N}$ aggiungendo anche un solo elemento. Altrettanto facile vedere che non ogni catena in F ha un maggiorante in F: per esempio la catena

$$\{0, 1, 2\} \subseteq \{0, 1, 2, 3\} \subseteq \{0, 1, 2, 3, 4\} \subseteq \ldots$$

non ha un maggiorante in F. Un tale elemento dovrebbe essere un sottinsieme finito di F che contiene tutti gli elementi della catena, ma questi sono arbitrariamente grandi. Si vede anche facilmente che un maggiorante della catena in F dovrebbe contenere l'unione di tutti gli elementi della catena, ossia $\bigcup_{i \in \mathbf{N}} \{0, 1, 2, \ldots, i\}$, ma questa unione è $\mathbf{N}$ e non è contenuta in nessun elemento di F.

Passiamo ora alla dimostrazione del Teorema di Compattezza utilizzando il Lemma di Zorn.

Teorema 1.5 (Teorema di Compattezza per linguaggi arbitrari)**.** *Sia T in un linguaggio proposizionale arbitrario. Se T è finitamente soddisfacibile allora T è soddisfacibile.*

Dimostrazione. Dimostriamo il verso non banale. Supponiamo dunque che T sia finitamente soddisfacibile. Consideriamo la famiglia delle teorie che estendono T (ossia della T' tali che $T \subseteq T'$ e sono finitamente soddifacibili. Sia $\mathcal{T}$ questa famiglia. $\mathcal{T}$ è non vuoto perché contiene almeno T'.

Ovviamente $(\mathcal{T}, \subseteq)$ è un ordine parziale, ossia gode delle proprietà di riflessività, antisimmetria e transitività.

Vogliamo ora verificare che $(\mathcal{T}, \subseteq)$ soddisfa le condizioni per applicare il Lemma di Zorn

Sia $C = (T_i \ : \ i \in I)$ una tale catena crescente, dove I è un insieme di indici. Ovviamente $\bigcup_{i \in I} T_i$ è un maggiorante della catena.

Basta dimostrare che $\bigcup_{i \in I} T_i$ è in $\mathcal{T}$, ossia che estende T e che è finitamente soddisfacibile. La prima proprietà è ovvia.

Consideriamo quindi un arbitrario sottinsieme finito di $\bigcup_{i \in I} T_i$, sia $X = \{F_1, \ldots, F_n\}$. Ogni proposizione F_i è elemento di un qualche elemento della catena $T_{t(i)}$ e dunque X è sottinsieme di un elemento della catena ed è pertanto finitamente soddisfacibile.

Applicando il Lemma di Zorn otteniamo che $\mathcal{T}$ contiene un massimale, ossia una teoria T_M tale che:

- $T_M \supseteq T$.
- T_M è finitamente soddisfacibile.
- Per ogni $T' \in \mathcal{T}$ se $T_M \subseteq T'$ allora $T_M = T'$.

Un tale elemento massimale gode di alcune interessanti proprietà.

(1) Sia F un elemento di T_M. Allora $\neg F$ non può appartenere a T_M, altrimenti $\{F, \neg F\}$ violerebbe il fatto che T_M è finitamente soddisfacibile.

(2) Sia F non in T_M, risulta che necessariamente $\neg F$ deve essere in T_M. Altrimenti T_M potrebbe estendersi con F oppure con $\neg F$ senza perdere la proprietà di essere finitamente soddisfacibile e ciò violerebbe la massimalità di T_M.

(3) Se F è un elemento di T_M e G è conseguenza logica di F allora anche G è in T_M. Se fosse infatti G non in T_M, allora per quanto dimostrato sopra avremmo $\neg G \in T_M$. Ma allora $\{F, \neg G\}$ contraddirrebbe il fatto che T_M è finitamente soddisfacibile.

Le proprietà di T_M appena dimostrate sono sufficienti a dimostrare che T_M possiede un modello. In un certo senso, T_M è già un modello (di se stessa).

Assegnamo i valori di verità mettendo a vero *tutte e sole* le variabili in T_M:

$$\alpha(p_i) = 1 \text{ se e solo se } p_i \in T_M.$$

Si osservi che la definizione è ben posta perché per ogni $p_i \in \mathcal{L}$ abbiamo che $p_i \in T_M$ o $\neg p_i \in T_M$ ma non entrambi – per le proprietà di T_M viste sopra.

Dimostrare che α soddisfa T_M è sufficiente a dimostrare che soddisfa anche T. Otteniamo il risultato dimostrando che vale una proprietà più forte:

$$\alpha(A) = 1 \text{ se e solo se } A \in T_M.$$

Procediamo per induzione sulla complessità della formula A. Il caso atomico è ovvio per definizione. Consideriamo uno a uno gli altri casi.

Caso 1: $A = \neg F$. $\neg F \in T_M$ se e solo se $F \notin T_M$ se e solo se $\alpha(F) = 0$ (per ipotesi induttiva) se e solo se $\alpha(\neg F) = 1$ (per definizione).

Caso 2: $A = F \wedge G$. Cominciamo osservando che $F \wedge G$ è in S_M se e solo se $F \in T_M$ e $G \in T_M$. Infatti, se $F \wedge G \in T_M$, dato che $F \wedge G \models F$ e $F \wedge G \models G$, dalle proprietà di T_M viste sopra abbiamo $F, G \in T_M$. Viceversa se $F \in T_M$ e $G \in T_M$ ma $F \wedge G \notin T_M$ abbiamo $\neg(F \wedge G) \in T_M$ (per le proprietà di T_M). In questo caso $\{F, G, \neg(F \wedge G)\}$ sarebbe un sottinsieme finito di T_M insoddisfacibile. Questo dimostra l'osservazione.

Dunque abbiamo $F \wedge G \in T_M$ se e solo se $F \in T_M$ e $G \in T_M$ se e solo se $\alpha(F) = 1$ e $\alpha(G) = 1$ (ipotesi induttiva) se e solo se $\alpha(F \wedge G) = 1$ (definizione).

I casi restanti ($A = F \vee G$ e $A = (F \to G)$) sono lasciati per Esercizio.

□

2. Conclusione e riformulazione

Abbiamo dimostrato il Teorema di Compattezza nella forma seguente.

Teorema 2.1. *Una teoria T è soddisfacibile se e solo se T è finitamente soddisfacibile.*

Concludiamo osservando l'equivalenza seguente.

Proposizione 2.2. *I due punti seguenti sono equivalenti:*

(1) *$T \models A$ se e solo se esiste un sottinsieme finito T_0 di T tale che $T_0 \models A$.*
(2) *T è soddisfacibile se e solo se T è finitamente soddifacibile.*

Dimostrazione. Dimostriamo il punto 2 dal punto 1. Sia T finitamente soddisfacibile ma non soddisfacibile. Se T è insoddisfacibile allora esiste A tale che $T \models A$ e $T \models \neg A$. Dall'ipotesi (1) abbiamo che esistono $B_1, \ldots, B_n \in T$ tali che $B_1, \ldots, B_n \models A$ ed esistono $C_1, \ldots, C_m \in T$ tali che $C_1, \ldots, C_m \models \neg A$. Ma allora $\{B_1, \ldots, B_n, C_1, \ldots, C_m\}$ è un sottinsieme finito di T insoddisfacibile, contro l'ipotesi.

Dimostriamo il punto (1) dal punto (2). (Esercizio).

□

LOGICA MATEMATICA

A.A. 23/24, DISPENSA N. 7

Sommario. Applicazioni matematiche del Teorema di Compattezza proposizionale.

1. Applicazioni della Compattezza

Esercizio 1.1. Sia $T = \{A_0, A_1, A_2, A_3, \dots\}$ una teoria proposizionale infinita numerabile. Supponiamo che ogni assegnamento soddisfa qualche proposizione in T (ossia: per ogni α esiste $i \in \mathbf{N}$ tale che $\alpha(A_i) = 1$. Dimostrare, usando il Teorema di Compattezza, che esiste un $j \in \mathbf{N}$ tale che $A_0 \vee A_1 \vee \dots \vee A_j$ è una tautologia (ossia: è soddisfatta da tutti gli assegnamenti).

Ragioniamo per assurdo, assumendo che (1) ogni assegnamento soddisfa qualche proposizione in T, ma (2) nessuna proposizione $A_0 \vee A_1 \vee \dots \vee A_j$ è una tautologia. Da (2) segue che per ogni $j \in \mathbf{N}$ esiste α tale che $\alpha(A_0 \vee A_1 \vee \dots \vee A_j) = 0$. Questo accade solo se $\alpha(A_0) = \alpha(A_1) = \dots \alpha(A_j) = 0$, che vale se e solo se $\alpha(\neg A_0) = \alpha(\neg A_1) = \dots \alpha(\neg A_j) = 1$, che possiamo riformulare dicendo che la teoria $\{\neg A_0, \neg A_1, \dots, \neg A_j\}$ è soddisfatta da α. Ricapitolando, l'assunzione (2) dice che per ogni $j \in \mathbf{N}$ la teoria $\{\neg A_0, \neg A_1, \dots, \neg A_j\}$ è soddisfacibile.

Da questo segue facilmente che la teoria $T^\neg = \{\neg A_0, \neg A_1, \neg A_2, \dots\}$ (ottenuta da T negando tutti i suoi elementi) è finitamente soddisfacibile. Per il Teorema di Compattezza questo implica che $T^\neg$ è soddisfacibile, ossia che esiste un α tale che $\alpha(T^\neg) = 1$, ossia esiste α tale che per ogni $i \in \mathbf{N}$, $\alpha(\neg A_i) = 1$, ossia esiste α tale che per ogni $i \in \mathbf{N}$, $\alpha(A_i) = 0$. Ma quest'ultimo fatto contraddice l'ipotesi (1).

Il Teorema di Compattezza permette di dare dimostrazioni piuttosto eleganti di alcuni teoremi non banali in diverse aree della Matematica.

1.1. Colorabilità di grafi.

Un grafo è k-colorabile se esiste una partizione dei vertici di G in k classi tali che non esistono archi tra i vertici di una stessa classe. In altre parole: è possibile colorare i vertici di G con k colori in modo tale che se due vertici sono connessi da un arco, allora hanno colori diversi.

Per formulare il Teorema ci serve la semplice nozione di sottografo indotto in un grafo da un insieme di vertici. Dato un grafo $G = (V, E)$ e un insieme di veritici $W \subseteq V$, il sottografo di G indotto da W è il grafo che ha come vertici W e come relazione d'arco la restrizione della relazione d'arco E di G a $W \times W$.

Teorema 1.2 (Edős-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.*

Si noti che nel caso di un grafo G finito il teorema resta banalmente vero. La dimostrazione originale è in N.G. De Bruijn, P. Erdős, *A colour problem for infinite graphs and a problem in the theory of relations*, **Nederl. Akad. Wetensch. Proc. Ser. A**, 54:371–37.

Dimostriamo il Teorema usando la Compattezza proposizionale.

Dimostrazione. Abbiamo già visto che la seguente teoria T cattura la nozione di k-colorazione per un grafo $G = (V, E)$.

(1) $P_{v,1} \vee P_{v,2} \vee \dots \vee P_{v,k}$, per ogni $v \in V$.
(2) $\neg(P_{v,i} \wedge P_{v,j})$ per ogni $v \in V$, $i \neq j$ in $[1, k]$.
(3) $\neg(P_{v,i} \wedge P_{w,i})$ per ogni $\{v, w\} \in E$, $i \in [1, k]$.

Note preparate da Lorenzo Carlucci, lorenzo.carlucci@uniroma1.it.

Abbiamo già osservato che se un assegnamento α soddisfa T allora la seguente colorazione di G in k colori è una k-colorazione di G:

$$c(v) = i \quad \text{se e solo se } \alpha(P_{v_i}) = 1.$$

Per dimostrare il teorema della k-colorabilità basta allora dimostrare che T sia soddisfacibile. Per Compattezza basta dimostrare che ogni sottinsieme finito di T è soddisfacibile.

Sia $T_0 \subseteq T$ finito. Consideriamo l'insieme V_0 dei vertici $v \in V$ menzionati in T_0, ossia tali che $P_{v,i}$ appare in qualche formula di T_0, per qualche $i \in [1, k]$. Per ipotesi, il sottografo di G indotto da questo insieme di vertici è k-colorabile. Sia c una tale k-colorazione. Possiamo usarla per definire un assegnamento che soddisfa T_0. Poniamo

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

altrimenti poniamo $\alpha(P_{v,i}) = 0$.

Si noti che T_0 può contenere una arbitraria combinazione finita di proposizioni dei tipi (1), (2), (3) che definiscono la teoria T (T_0 può essere anche vuota). Possiamo però asserire che T_0 è contenuta nella teoria T_0^* che contiene tutti gli assiomi di T che coinvolgono soltanto i vertici menzionati in T_0. Ovviamente, se T_0^* è soddisfatta da un assegnamento, anche T_0 è soddisfatta dallo stesso assegnamento.

Si dimostra facilmente che $\alpha(T_0^*) = 1$ (dettagli lasciati per esercizio), dove α è l'assegnamento definito sopra, indotto dalla colorazione del grafo indotto dai vertici menzionati in T_0. Dunque $\alpha(T_0) = 1$.

Dunque per Compattezza T è soddisfacibile. Come già osservato un assegnamento α che soddisfa T induce naturalmente una k-colorazione $c : V \to [1, k]$ di G, ponendo

$$c(v) = i \text{ se e soltanto se } \alpha(P_{v,i}) = 1.$$

□

1.2. **Estensioni totali.** Usando il teorema di Compattezza si può dimostrare il seguente teorema sull'estendibilità di ogni ordine parziale a un ordine totale, assumendo che valga la sua versione finitaria.

Teorema 1.3. *Per ogni ordine parziale $(X, <)$ esiste un ordine totale $(X, \prec)$ tale che $\prec$ estende $<$, ossia per ogni $x, y \in X$, se $x < y$ allora $x \prec y$.*

Dimostrazione. Abbiamo già osservato che la seguente teoria proposizionale cattura la nozione di ordine totale stretto.

(1) $\neg p_{x,x}$ per ogni $x \in X$.
(2) $p_{x,y} \to \neg p_{y,x}$ per ogni $x, y \in X$.
(3) $(p_{x,y} \wedge p_{y,z}) \to p_{x,z}$ per ogni $x, y, z \in X$
(4) $p_{x,y} \vee p_{y,x}$ per ogni $x, y \in X$ con $x \neq y$.

Se α è un assegnamento allora la relazione $\prec_\alpha$ su X definita come segue

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

è tale che $\alpha(T) = 1$ se e solo se $\prec_\alpha$ è un ordine totale stretto su X.

Consideriamo l'estensione della teoria vista sopra con le formule seguenti:

(5) $p_{x,y}$ per ogni $x, y \in X$ con $x < y$.

Queste formule esprimono il fatto che l'ordine totale che stiamo descrivendo estende l'ordine di partenza $<$ su X. Chiamiamo T questa teoria (si noti che dipende da $(X, <)$). Sappiamo già che ogni modello di T induce un ordine totale su X perché T contiene le proposizioni dei gruppi 1-4. Un modello di T è dunque tutto ciò che ci serve, perché l'ordine indotto da un assegnamento che soddisfa questa teoria sarà un ordine totale che estende $<$ grazie alla presenza del gruppo (5).

Per Compattezza è sufficiente dimostrare che ogni sottinsieme T_0 finito di T è soddisfacibile. In un tale sottinsieme finito viene menzionato un insieme finito di elementi di X, sia esso X_0, composto dagli $x, y \in X$ tali che la variabile $p_{x,y}$ compare nel sottinsieme finito T_0 di T. Consideriamo la restrizione di $<$ a X_0. Si tratta ovviamente di un ordine parziale finito. A questo punto dobbiamo essere in grado di dimostrare che T_0 è soddisfacibile. A questo scopo sarebbe sufficiente esibire un ordine totale su X_0 che estende $<$ su X_0. Non ci resta che dimostrare che ogni ordine parziale finito ammette una estensione totale. Questo asserto non è altro che la *versione finita* del teorema che stiamo dimostrando. In questo caso, va dimostrata a parte (la dimostrazione è per esercizio – si ottiene facilmente per induzione sulla cardinalità di X_0). Sia dunque

$\prec_0$ un ordine totale su X_0 che estende $<$. Si osserva facilmente che un tale ordine induce un assegnamento α_0 che soddisfa T_0, ponendo

$$\alpha_0(p_{x,y}) = 1 \text{ se e solo se } x, y \in X_0 \text{ e } x < y.$$

Per Compattezza concludiamo che T è soddisfacibile. Sia α un assegnamento che soddisfa T. Basta porre $v(p_{x,y})$ a 1 sse $x \prec y$. Dunque esiste un ordine totale $\prec$ su X che estende $<$. □

2. Lemma di Kőnig

Un *albero binario* è un sottinsieme di stringhe binarie finite chiuso per segmento iniziale (denotiamo la relazione di segmento iniziale con $\preceq$). Un *ramo infinito* di un albero T è, intuitivamente, una successione di stringhe di T di ogni possibile lunghezza, o più formalmente un sottinsieme di T totalmente ordinato e massimale. In altri termini un ramo infinito è una funzione $r : \mathbf{N} \to \{0, 1\}$ tale che r ristretta a $[n]$ è in T per ogni n (più precisamente la stringa binaria $(r(0), r(1), \ldots, r(n)) \in T$).

Il seguente Teorema è noto come Lemma di Kőnig (per alberi binari).

Teorema 2.1 (Lemma di Kőnig per alberi binari)**.** *Ogni albero binario infinito contiene un ramo infinito.*

Più in generale, un *albero* è un ordine parziale con una unica radice e tale che per ogni nodo dell'albero l'insieme dei suoi predecessori è finito e linearmente ordinato. Il lemma di Kőnig vale anche per una classe di alberi più grande di quelli binari, ossia gli alberi *a ramificazione finita*, ossia tali che ogni nodo dell'albero ha un numero finito di successori immediati.

Teorema 2.2 (Lemma di Kőnig generale)**.** *Ogni albero infinito a ramificazione finita ha un ramo infinito.*

Dettagliamo una dimostrazione del Lemma di Kőnig binario usando la Compattezza.

Dimostrazione. Sia $(A, <_A)$ un albero binario infinito. Per ogni n consideriamo l'insieme A_n degli elementi di A di livello (lunghezza) n.

Dato che ogni A_n è finito ma A è infinito, per infiniti n deve valere che A_n non è vuoto.

Inoltre se A_n è non vuoto e $m < n$ allora A_m è non vuoto.

Dunque per ogni n esistono elementi di A di lunghezza n. Si noti bene che questo non significa che A coincide con l'albero binario completo in cui tutti i livelli sono pieni (ossia hanno tutti i possibili 2^n elementi)!

Vogliamo ora definire una teoria che ci permetta di ottenere il risultato usando la Compattezza. Dato che il nostro scopo è di ottenere un ramo infinito nell'albero A, definiamo una teoria che descrive l'esistenza di un ramo infinito in A.

Un ramo infinito in A è un insieme R di nodi di A ordinato linearmente relativamente ad $<_A$ tale che ha un unico nodo a ogni livello.

Fissiamo il linguaggio proposizionale $\mathcal{L}$ contenente una variabile p_s per ogni stringa binaria finita s. Consideriamo la teoria T_A composta dalle seguenti formule.

(1) Per ogni $n \in \mathbf{N}$, se $\{\sigma_1, \ldots, \sigma_t\}$ sono i nodi di livello n in A:

$$p_{\sigma_1} \vee p_{\sigma_2} \vee \cdots \vee p_{\sigma_t}.$$

(2) Per ogni $n \in \mathbf{N}$, per ogni $\sigma, \tau \in A$ di livello n con $\sigma \neq \tau$:

$$\neg(p_\sigma \wedge p_\tau).$$

(3) Per ogni $\sigma, \tau \in A$ con $\tau <_A \sigma$:

$$p_\sigma \to p_\tau.$$

Intuitivamente le formule della teoria T_A asseriscono che: per ogni livello possibile viene inclusa una stringa di quel livello; per ogni livello possibile viene inclusa al più una stringa di quel livello; e che se una stringa viene inclusa allora vengono incluse anche tutti i suoi segmenti iniziali.

Cominciamo col dimostrare che T_A è scelta in modo adeguato: se T_A è soddisfacibile allora posso concludere che in A esiste un ramo infinito. Sia α un assegnamento che soddisfa T_A.

Mostriamo come estrarre un ramo infinito in A da α.

Per ogni $n > 0$ esiste un'unica $s \in A$ di livello n tale che $\alpha(p_s) = 1$. Ne esiste almeno una perché T_A contiene le proposizioni di tipo (1): dato n, siano $\sigma_1, \ldots, \sigma_t$ le stringhe in A di livello n. T_A contiene

$p_{\sigma_1} \vee \cdots \vee p_{\sigma_t}$ e dunque $\alpha(p_{\sigma_1} \vee \cdots \vee p_{\sigma_t}) = 1$, il che implica che esiste un $i \in [1, t]$ tale che $\alpha(p_{\sigma_i}) = 1$. L'unicità segue dal fatto che T_A soddisfa le proposizioni di tipo (2): siano n e σ, τ tali che σ e τ sono entrambe di livello n e $\alpha(p_\sigma) = 1 = \alpha(p_\tau)$. Questo contraddice il fatto che $\alpha(\neg(p_\sigma \wedge p_\tau)) = 1$.

Per ogni n sia σ_n l'unica stringa σ di livello n in A tale che $\alpha(p_\sigma) = 1$.

Se $m \leq n$ allora $\sigma_m <_A \sigma_n$. Da una parte, T contiene tutte le proposizioni $p_{\sigma_n} \to p_\tau$ per ogni $\tau \preceq \sigma_n$. Dall'altra, esiste necessariamente un segmento iniziale di σ_n nell'albero al livello m. Sia questo elemento τ. Se τ è σ_m abbiamo concluso. Altrimenti, dato che $\alpha(p_{\sigma_n} \to p_\tau) = 1$ per definizione di T e scelta di α e $\alpha(p_{\sigma_n}) = 1$ per scelta di σ_n, deve essere $\alpha(p_\tau) = 1$. Ma per definizione di T e scelta di α deve anche valere $\alpha(\neg(p_{\sigma_m} \wedge p_\tau)) = 1$, perché τ e σ_m sono allo stesso livello (ossia m). Ma questo significa che $\alpha(p_{\sigma_m}) = 0$, contro la scelta di σ_m.

Ponendo $R = \{s \in 2^{<\omega} : \alpha(p_s) = 1\}$ abbiamo un ramo infinito per le osservazioni di sopra: R interseca A a ogni livello e lo interseca in un unico punto (ed è dunque infinito) e $<_A$ su R è un ordine lineare. Dimostriamo per Compattezza che T_A è soddisfacibile dimostando che è finitamente soddisfacibile. Sia T_0 un sottinsieme finito di T_A. Siano $\sigma_1, \ldots, \sigma_i$ tutte e sole le stringhe binarie menzionate in T_0 (ossia tali che la variabile associata compaia in T_0). Sia n la lunghezza massima di tali stringhe e sia s_m di livello n. Definiamo il nostro assegnamento inducendolo dalla struttura del ramo finito di A che inizia dalla radice e termina in s_m. Definiamo α ponendo

$$\alpha(p_\sigma) = 1 \text{ se e solo se } \sigma <_A \sigma_m.$$

Si verifica facilmente che per ogni formula $F \in T_0$ abbiamo $\alpha(F) = 1$. Dunque per Compattezza T_A è soddisfacibile.

□

Anche la forma generale del Lemma di Kőonig (per alberi infiniti a ramificazione finita) può essere dimostrata per Compattezza (Esercizio).

3. Da Kőnig alla Compattezza (Esercizio guidato)

Vale anche l'implicazione inversa ossia si può dimostrare il Teorema di Compattezza proposizionale per linguaggi numerabili usando il Lemma di Kőnig.

Dimostrazione. Sia T una teoria in un linguaggio numerabile $\mathcal{L} = \{p_1, p_2, p_3, \ldots\}$. Sia T finitamente soddisfacibile. Vogliamo dimostrare che T è soddisfacibile, usando il Lemma di Kőnig.

Questo ci permette di estrarre un ramo infinito da un albero binario infinito dunque iniziamo cercando di costruire un albero infinito cui poter applicare il Lemma.

I nostri dati di partenza sono assegnamenti che soddisfano porzioni finite di T e il nostro scopo è di ottenere un assegnamento che soddisfa tutta T. Possiamo concettualizzare il problema come quello di definire un assegnamento di questo tipo in base a sue *approssimazioni finite*.

Definiamo un albero A i cui elementi sono *assegnamenti parziali* ossia funzioni finite di tipo $\{p_1, \ldots, p_n\} \to \{0, 1\}$. Di questi ci interesseranno soltanto quelli che soddisfano tutte le proposizioni di T in un cui compaionio soltanto le variabili proposizionali $\{p_1, p_2, \ldots, p_n\}$.

- L'insieme delle proposizioni in T in cui compaiono soltanto le variabili $\{p_1, \ldots, p_n\}$ è necessariamente finito?
- Se ho un assegnamento che soddisfa, per ogni n, tutte le proposizioni di T nelle variabili $\{p_1, \ldots, p_n\}$, cosa posso concludere?

Definiamo il nostro albero A per livelli. Il livello 0 è vuoto. Il livello $n > 1$ contiene tutti e soli gli assegnamenti parziali $\alpha : \{p_1, p_2, \ldots, p_n\} \to \{0, 1\}$ che soddisfano tutte le formule in T in cui vengono menzionate solo le variabili $\{p_1, \ldots, p_n\}$.

L'ordine $<_A$ che consideriamo sui nodi di A è il seguente: se α è di livello n (ossia di tipo $\alpha : \{p_1, \ldots, p_n\} \to \{0, 1\}$) e β è di livello m (ossia di tipo $\beta : \{p_1, \ldots, p_m\} \to \{0, 1\}$) con $n < m$, poniamo $\alpha <_A \beta$ se e solo se β ristretto a $\{p_1, \ldots, p_n\}$ coincide con α. In altre parole, ordiniamo gli elementi di A per restrizione.

Verificare che:

- A con l'ordine definito è un albero
- Ogni livello di A è finito

Per applicare il Lemma di Kőnig dobbiamo assicurarci che A sia infinito.

Facciamo vedere che nessun livello di A è vuoto. Fissiamo $n > 1$ (il caso $n = 0$ è banale: perché?). Consideriamo l'insieme delle formula di T in cui compaiono solo le variabili $\{p_1, \ldots, p_n\}$, sia T_n questo insieme. Ragioniamo per casi.

Caso 1: T_n è finito. Completare la dimostrazione

Caso 2: T_n è infinito. Completare la dimostrazione

Sia

$$T_n = \{F_1, F_2, F_3, \ldots\}$$

dove le F_i sono tutte e sole le formule di T in cui compaiono soltanto le variabili $\{p_1, \ldots, p_n\}$.

T_n può essere ovviamente ottenuto come unione delle sue seguenti approssimazioni finite:

$$I_1 := \{F_1\}, I_2 = \{F_1, F_2\}, I_3 = \{F_1, F_2, F_3\}, \ldots, I_k = \{F_1, F_2, \ldots, F_k\}, \ldots$$

Perché sono sicuro che per ogni k esiste un assegnamento α_k che soddisfa I_k?

Dimostrare che esiste un assegnamento $\alpha : \{p_1, \ldots, p_n\} \to \{0, 1\}$ tale che per infiniti k si ha $\alpha = \alpha_k$.

Concludere.

QED.

L'albero A che abbiamo definito soddisfa le ipotesi del Lemma di Kőnig. Dunque contiene almeno un ramo infinito, sia R. R è infinito e interseca ogni livello di A in unico punto ed è linearmente ordinato. Dunque R determina una sequenza infinita

$$\alpha_0 \subseteq \alpha_1 \subseteq \alpha_2 \subseteq \ldots$$

di assegnamenti parziali dove $\alpha_i : \{p_1, \ldots, p_i\} \to \{0, 1\}$ soddisfa tutte le formula di T nelle variabili $\{p_1, \ldots, p_n\}$ e $\alpha_i \subset \alpha_{i+1}$.

Definire un α che soddisfa T.

□

LOGICA MATEMATICA

A.A. 23/24, DISPENSA N. 8

SOMMARIO. Applicazioni logiche e algoritmiche della Compattezza. Semi-decidibilità e decidibilità delle conseguenze logiche di una teoria.

1. Compattezza e decidibilità algoritmica

Visto il potere espressivo della Logica Proposizionale e la relazione tra proprietà interessanti di strutture matematiche e la nozione di soddisfacibilità o conseguenza logica, è naturale chiedersi se sia possibile *automatizzare* la risposta alla domanda:

$$T \models A?$$

per una generica teoria proposizionale T e una formula A. Questa domanda in particolare ha dato – storicamente – gran parte dell'impeto allo sviluppo della logica matematica moderna.

Osservazione 1.1. Ragioniamo qui in termini di una nozione *informale* di procedura meccanica, o *algoritmo*. Ci poniamo cioè nella stessa situazione dei matematici che consideravano per la prima volta i sistemi di logica formale che stiamo studiando: l'idea di algoritmo era per loro rappresentata da una successione di regole non ambigue applicabili – in teoria – da una macchina; anche se non esistevano vere e proprie macchine calcolatrici. Si pensi, come modello di riferimento, all'algoritmo della divisione euclidea. Vedremo più avanti una definizione rigorosa di algoritmo ma per la discussione attuale non ne abbiamo bisogno!

Se T è una **teoria finita**, sia $T = \{A_1, \dots, A_n\}$ sappiamo già rispondere: $T \models A$ equivale infatti a $\models (A_1 \wedge \dots \wedge A_n) \to A$ (o $\models (A_1 \to (A_2 \to \dots (A_{n-1} \to A_n) \dots)$, che equivale a dire che la singola proposizione $(A_1 \wedge \dots \wedge A_n) \to A$ è una tautologia. Ovviamente esiste un metodo *automatico* o *automatizzabile* per verificare quest'ultimo fatto: basta costruire la tavola di verità e controllare se l'ultima colonna è tutta di 1. Se questo è il caso, si risponde affermativamente e altrimenti si risponde negativamente.

Cosa possiamo dire se T è una teoria **infinita numerabile**, scritta in un linguaggio numerabile $\mathcal{L} = \{p_1, p_2, \dots\}$? La definizione della relazione $T \models A$ è fortemente infinitaria, e prevede una quantificazione universale su uno spazio non-numerabile (tutti gli assegnamenti sul linguaggio infinito di T). Dal Teorema di Compattezza abbiamo però l'equivalenza seguente:

(1) $T \models A$
(2) Esiste un sottinsieme finito T_0 di T tale che $T_0 \models A$.

Indichiamo con $Fin(T)$ l'insieme dei sottinsiemi finiti di T. Se T è numerabile anche $Fin(T)$ è numerabile. Immaginiamo di saper produrre una lista (o enumerazione) di tutti e soli i sottinsiemi finiti di T, sia questa lista

$$S_1, S_2, S_3, \dots$$

Dal Teorema di Compattezza sappiamo che $T \models A$ se e solo se esiste un $i \in \mathbf{N}$ tale che $S_i \models A$.

Possiamo immaginare allora la seguente **procedura** meccanica/algoritmica/effettiva:

> Partiamo dal primo elemento della lista e chiediamoci se $S_1 \models A$. Dato che S_1 è finito a questa domanda so rispondere algoritmicamente. Se la risposta è sì, terminiamo la procedura e rispondiamo sì alla domanda iniziale. Altrimenti procediamo e consideriamo $S_2 \models A$? Se sì terminiamo e rispondiamo sì, altrimenti consideriamo $S_3 \models A$ e così via.

Note preparate da Lorenzo Carlucci, lorenzo.carlucci@uniroma1.it.

Se l'enumerazione di $Fin(T)$ si può produrre meccanicamente, allora l'intera procedura è mecccanica. Cosa garantisce?

Se veramente $T \models A$ allora esiste un i tale che $S_i \models A$ e in un tempo finito la mia procedura lo trova, verifica la relazione e termina rispondendo sì.

Se invece $T \not\models A$ allora non ho garanzie che la mia procedura termini entro un numero finito di passi. Anzi, sono sicuro che la procedura continuerà indefinitamente. In questo caso diciamo che diverge o è indefinita.

Quando un problema ammette una soluzione di questo tipo diciamo che è **semi-decidibile**. Dunque, quanto appena visto si può riassumere così:

Se T è una teoria infinita in un linguaggio numerabile tale che esiste una enumerazione algoritmica di tutti e soli i suoi sottinsiemi finiti allora il problema $T \models A$ è semi-decidibile.

Risulta naturale chiedersi a quali condizioni possiamo essere certi che si possano enumerare automaticamente tutti e soli i sottinsiemi fintii di T. Si osserva piuttosto facilmente che questo è certamente possibile se T stessa (intesa come insieme infinito di proposizioni) può essere enumerata automaticamente. In altre parole: se ho una procedura meccanica per produrre una lista infinita

$$L_1, L_2, L_3, \dots$$

di tutte e sole le proposizioni di T allora ho una procedura meccanica per produrre una lista infinita

$$S_1, S_2, S_3, \dots$$

di tutti e soli i sottinsiemi finiti di T.

Un insieme infinito per cui esiste una procedura meccanica di enumerazione di tutti e soli i suoi elementi è detto **computabilmente enumerabile** (o ricorsivamente enumerabile, o algoritmicamente enumerabile, o effettivamente enumerabile). Riassumiamo le considerazioni di sopra come segue:

Proposizione 1.2. *Se T è computabilmente enumerabile allora il problema della conseguenza logica da T ($T \models A$?) è semi-decidibile.*

Osservazione 1.3. Potrebbe risultare difficile concepire l'esistenza di una teoria numerabile $T = \{A_1, A_2, \dots\}$ ma non computabilmente enumerabile. Come primo approccio un argomento di cardinalità può risultare utile. Una teoria computabilmente enumerabile è tale che esiste una procedura meccanica (algoritmica) capace di produrre una lista di tutte e sole le proposizioni della teoria. Una tale procedura meccanica può essere equiparata a un programma in un linguaggio di programmazione. Poiché i programmi consistono in una sequenza *finita* di istruzioni (in un linguaggio formale numerabile!), il numero dei possibili programmi che possiamo scrivere in un linguaggio di programmazione dato è infinito numerabile. D'altra parte, il numero di teorie numerabili in un linguaggio proposizionale è più che numerabile, in quanto corrisponde a tutti i possibili sottinsiemi numerabili dell'insieme delle proposizioni. Dunque necessariamente esistono teorie numerabili che non sono computabilmente enumerabili. Vedremo più avanti esempi più "concreti" di teorie di questo genere.

La semi-decidibilità di una teoria T è un risultato interessante: ci garantisce l'esistenza di un algoritmo che risponde correttamente a tutte le domande di tipo $T \models A$? che ammettono una risposta affermativa. Ossia: se A consegue logicamente da T l'algoritmo lo scopre in un tempo finito e ce lo segnala.

Si osservi che se T è finita abbiamo molto di più: esiste una procedura effettiva (un algoritmo) tale che, per ogni A, se $T \models A$ allora l'algoritmo termina e risponde sì e se $T \not\models A$ allora l'algoritmo termina e risponde no.

In questo caso diciamo che il problema della conseguenza logica da T è (algoritmicamente) **decidibile** (fino al sì e al no).

Risulta naturale chiedersi a che condizioni su T possiamo garantire l'esistenza di un algoritmo di questo tipo. Per rispondere cerchiamo di trarre vantaggio da quanto già sappiamo: se T è computabilmente enumerabile, allora abbiamo una procedura di semi-decisione a ogni domanda di tipo $T \models A$?

Consideriamo la domanda duale $T \models \neg A$? Possiamo applicare a $\neg A$ la procedura di semi-decisione: se veramente $T \models \neg A$ allora la procedura termina e risponde affermativamente.

Naturalmente una teoria T per cui esiste una A tale che $T \models A$ e $T \models \neg A$ è una teoria inutile (abbiamo già dimostrato che è insoddisfacibile). Possiamo quindi ragionevolmente escludere a priori questo caso.

Stiamo considerando la seguente **procedura** effettiva:

Sia T una teoria computabilmente enumerabile e sia $S_1, S_2, S_3, \ldots$ una enumerazione meccanica di tutti i suoi sottinsiemi finiti. Data una proposizione A nel linguaggio di T, procediamo come segue: Controlliamo in sequenza:

$$S_1 \models A?, S_1 \models \neg A?, S_2 \models A?, S_2 \models \neg A?, \ldots$$

Se $S_i \models A$ terminiamo e rispondiamo 'sì'. Se $S_i \models \neg A$ terminiamo e rispondiamo 'no'.

Cosa ci garantisce questa procedura? Abbiamo già escluso il caso in cui $T \models A$ e $T \models \neg A$. Consideriamo i casi restanti.

Caso 1. $T \models A$ e $T \not\models \neg A$: In questo caso la procedura di semi-decisione applicata a A termina e risponde affermativamente mentre la procedura applicata a $\neg A$ diverge. Possiamo comunque concludere con certezza che $T \models A$.

Caso 2. $T \not\models A$ e $T \models \neg A$: In questo caso la procedura di semi-decisione applicata a A diverge e la procedura applicata a $\neg A$ termina e risponde affermativamente. Possiamo comunque concludere con certezza che $T \models \neg A$. Se T non è insoddisfacibile, non può essere che $T \models A$. Dunque in questo caso possiamo concludere e rispondere negativamente alla domanda iniziale: $T \not\models A$.

Caso 3. $T \not\models A$ e $T \not\models \neg A$. In questo caso la procedura diverge tanto quando la applichiamo a A che quando la applichiamo a $\neg A$. Entrambe le procedure continuano indefinitamente e non ci danno mai una risposta. Una teoria di questo tipo non è da escludere in linea di principio: se la escludiamo però, garantiamo che la procedura che stiamo descrivendo (valutare in parallelo se $T \models A$ e se $T \models \neg A$) non può mai trovarsi in questo caso indefinito.

Dalle considerazioni qui sopra è naturale isolare la seguente proprietà:

Definizione 1.4 (Completezza semantica di una teoria)**.** Una teoria T è (semanticamente) completa se per ogni A nel linguaggio di T vale esattamente una tra

$$T \models A \text{ e } T \models \neg A.$$

Si noti che nella definizione di teoria completa stiamo escludendo il caso di una teoria insoddisfacibile. Per una tale teoria vale infatti che $T \models A$ e anche $T \models \neg A$. Per questo motivo richiediamo che valga una e una sola di tali conseguenze logiche.

Ricapitolando la discussione vista sopra: se T è enumerabile e T è completa allora, per ogni A, posso applicare la procedura vista sopra. Se vale $T \models A$ sono sicuro di avere una risposta affermativa in tempo finito: esiste un i tale che $S_i \models A$ e prima o poi lo trovo. Posso allora concludere che $T \models A$. Se vale $T \models \neg A$ sono sicuro di avere una risposta affermativa in tempo finito alla domanda $T \models \neg A?$: esiste un i tale che $S_i \models \neg A$ e prima o poi lo trovo. e quindi di poter rispondere negativamente alla domanda inziiale: $T \not\models A$. Ho anche la garanzia di non trovare mai un i tale che $S_i \models A$ e un j tale che $S_j \models \neg A$ perché ho assunto che T non può implicare logicamente una proposizione e la sua negazione. Ho così descritto un algoritmo di decisione per il problema della conseguenza logica da T.

Proposizione 1.5. *Se T è una teoria enumerabile e completa allora la domanda $T \models A?$ è decidibile algoritmicamente, per ogni formula A.*

La Proposizione qui sopra individua nella completezza di una teoria enumerabile T una *condizione sufficiente* all'esistenza di una procedura automatica di decisione a tutte le domande del tipo $T \models A?$ Si tratta di un risultato importante: se riesco a formalizzare una certa branca della Matematica in una teoria T enumerabile e completa ho la garanzia a priori di poter meccanizzare/automatizzare tutte le verità (esprimibili nel linguaggio) riguardanti quella branca della Matematica.

Risulta dunque di un certo interesse capire cosa è possibile formalizzare con una teoria enumerabile e completa.

2. Osservazioni aggiuntive

La teoria definita durante la dimostrazione del Teorema di Compattezza ha la proprietà di essere massimale tra le teorie soddisfacibili che estendono la teoria di partenza. Vediamo qui sotto che questa proprietà coincide essenzialmente la completezza semantica. Se T è una teoria e A una proposizione, non è detto che valga uno tra $T \models A$ e $T \models \neg A$. Possono infatti esistere assegnamenti che soddisfano T e A e altri assegnamenti che soddisfano T e $\neg A$. Una teoria completa (semanticamente) è una teoria per cui questo non accade. Per ogni scenario (proposizione) possibile, una teoria completa ha l'uno o l'altro scenario come conseguenza logica.

Vale la seguente caratterizzazione.

Proposizione 2.1. *Le proprietà seguenti sono equivalenti.*

(1) *T è semanticamente completa.*
(2) *Per ogni formula A, vale $T \models A$ se e solo se $T \not\models \neg A$.*
(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 $T \not\models A \to B$ se e solo se $T \models A$ e $T \models \neg B$.*

Dimostrazione. Esercizio. □

Definizione 2.2. Denotiamo con $Con(T)$ l'insieme delle *conseguenze logiche della teoria* T, ossia poniamo

$$Con(T) = \{A \,:\, T \models A\}.$$

Una teoria è detta *semanticamente chiusa* se contiene tutte le sue conseguenze logiche (e dunque coincide con $Con(T)$. Altrimenti detto, se $T \models A$ allora $A \in T$.

In alcuni contesti è conveniente avere a che fare con teorie semanticamente chiuse (in alcuni libri di testo il concetto di teoria viene definito come insieme semanticamente chiuso). Per quanto riguarda la soddisfacibilità passare da una teoria T alla corrispondente teoria semanticamente chiusa $Con(T)$ non ha effetti:

Osservazione 2.3. T è soddisfacibile se e solo se $Con(T)$ è soddisfacibile (Esercizio).

Se v è un assegnamento allora l'insieme di tutte e sole le proposizioni soddisfatte da v è una teoria massimalmente soddisfacibile (Esercizio).

Una teoria semanticamente chiusa e completa ha l'interessante proprietà di essere massimale rispetto all'inclusione insiemistica relativamente alle teorie soddisfacibili.

Definizione 2.4. Diciamo che T è *massimalmente soddisfacibile* sse T è soddisfacibile e per ogni $T' \supseteq T$ se T' è soddisfacibile allora $T = T'$.

Proposizione 2.5. *T è semanticamente chiusa e completa se e solo se T è massimalmente soddisfacibile.*

Dimostrazione. Esercizio. □

LOGICA MATEMATICA

A.A. 23/24, DISPENSA N. 9

Sommario. Sistemi di deduzione formale. Teorema di Completezza proposizionale (enunciato e strategia di dimostrazione).

1. 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 o assioma o a una regola di ragionamento corretta che ci ha permesso di ottenerla da altre proposizioni.

Introduciamo qui sotto una formalizzazione rigorosa della nozione intuitiva di dimostrazione (per la logica proposizionale). L'interesse di questa formalizzazione non è solo concettuale (formalizziamo l'idea intuitiva di dimostrazione ipotetico-deduttiva) ma anche *algoritmico*.

Per semplificare le induzioni limitiamo il linguaggio ai connettivi $\neg, \to$. Gli altri connettivi si possono introdurre per definizione (come abbreviazioni).

Gli assiomi del Calcolo Proposizionale sono dati dai seguenti schemi (X, Y, Z variano su proposizioni).

Ax 1 $(X \to (Y \to X))$
Ax 2 $(X \to (Y \to Z) \to ((X \to Y) \to (X \to Z))$
Ax 3 $(\neg Y \to \neg X) \to (X \to Y)$

Gli assiomi qui sopra sono per la precisione *schemi di assioma*. Ogni formula ottenuta sostituendo X, Y, Z con proposizioni della logica proposizionale è una *istanza* di assioma. Questi assiomi rappresentano ipotesi di natura puramente logica che possiamo assumere come vere nelle nostre dimostrazioni. Si controlla facilmente che ogni istanza degli schemi di assioma qui sopra è una verità logica (tautologia).

Introduciamo una unica regola di inferenza, che formalmente è una relazione $R(X, Y, Z)$ tra tre proposizioni, dove X, Y sono le premesse e Z la conclusione e si dice che Z *si ottiene* da X, Y applicando la regola R, o che Z *segue* da X, Y per la regola R. La regola che usiamo è il cosidetto Modus Ponens: da X e $(X \to Y)$ segue Y.

Definizione 1.1 (Deduzione)**.** Una deduzione (o derivazione o prova) è una sequenza finita $F_1, \dots, F_k$ di proposizioni tale che per ogni $i \in \{1, \dots, k\}$,

- F_i è una istanza di un assioma, oppure
- F_i si ottiene da due formule precedenti nella sequenza per Modus Ponens, i.e., esistono $m, \ell < k$ tali che F_ℓ è $(F_m \to F_i)$.

Una formula F è un teorema del calcolo proposizionale se esiste una deduzione la cui ultima formula è F. In tal caso scriviamo $\vdash F$.

Esempio. $\vdash B \to B$. (Esercizio)

Si osserva facilmente che il calcolo appena definito è *corretto* nel senso che ogni teorema è una tautologia.

Note preparate da Lorenzo Carlucci, lorenzo.carlucci@uniroma1.it.

Proposizione 1.2 (Correttezza)**.** *Tutti i teoremi sono tautologie:*

$$\vdash B \Longrightarrow \models B.$$

Dimostrazione. Ogni istanza di un assioma è una tautologia, e il Modus Ponens conserva la verità: se A è vera e $A \to B$ è vera, allora B è vera. □

Esempio 1.3. Asserire $\vdash B \to B$ significa: Esiste una deduzione formale (dai soli assiomi logici) della formula $B \to B$.

Una possibile deduzione è la seguente:

$(B \to ((B \to B) \to B)) \to (((B \to (B \to B)) \to (B \to B))$
$B \to ((B \to B) \to B)$
$((B \to (B \to B)) \to (B \to B))$
$B \to (B \to B)$
$B \to B$
Esercizio: giustificare ogni riga.

Possiamo estendere la nozione di deduzione appena definita in modo da formalizzare una dimostrazione condotta in base a un insieme di ipotesi che non sono assiomi della logica.

Definizione 1.4 (Deduzione da premesse)**.** Se $\mathcal{S}$ è un insieme di proposizioni (anche infinito), e F è una formula, diciamo che F è deducibile dalle premesse $\mathcal{S}$ se esiste una sequenza finita di formula $F_1, \ldots, F_k$ tale che per ogni $i \in \{1, \ldots, k\}$,

- F_i è un assioma, oppure
- F_i è in $\mathcal{S}$, oppure
- F_i si ottiene per Modus Ponens da due formule che la precedono nella sequenza.

In tal caso scriviamo $\mathcal{S} \vdash F$.

Per semplicità, per un insieme $\mathcal{S}$ e formule $A_1, \ldots, A_n, B$, scriviamo $\mathcal{S}, A_1, \ldots, A_n \vdash B$ invece di usare la notazione più corretta $\mathcal{S} \cup \{A_1, \ldots, A_n\} \vdash B$ e analogamente se $\mathcal{S}_1, \ldots, \mathcal{S}_n$ sono insiemi e B è una formula scriviamo $\mathcal{S}_1, \ldots, \mathcal{S}_n \vdash B$ per $\mathcal{S}_1 \cup \cdots \cup \mathcal{S}_n \vdash B$. Ricordiamo che questo è sempre un asserto esistenziale, che dice: esiste una successione finita di formule con certe proprietà la cui ultima formula è F. La proposizione seguente è una conseguenza immediata delle definizioni.

Proposizione 1.5 (proprietà fondamentali di $\mathcal{S} \vdash F$)**.**

(1) *Se $\mathcal{S} \vdash F$ e $\mathcal{S} \subseteq \mathcal{S}'$ allora $\mathcal{S}' \vdash F$.*
(2) *$\mathcal{S} \vdash F$ se e solo se esiste un sottinsieme finito $\mathcal{U} \subseteq \mathcal{S}$ tale che $\mathcal{U} \vdash F$.*
(3) *Se $\mathcal{S} \vdash F$ e per ogni $A \in \mathcal{S}$ vale $\mathcal{U} \vdash A$, allora $\mathcal{U} \vdash F$.*

Osservazione 1.6. Il calcolo deduttivo di tipo assiomatico che abbiamo introdotto (formato da assiomi e regole di inferenza) è uno tra numerosi formalismi deduttivi e viene detto *alla Hilbert*. Si tratta di una definizione economica che ha lo svantaggio di non rispecchiare la struttura logica delle dimostrazioni naturali in Matematica. Esistono altri formalismi che meglio rispecchiano tale struttura e permettono l'analisi di interessanti proprietà generali della deduzione logica, per esempio la cosidetta Deduzione Naturale e il Calcolo dei Sequenti, entrambi dovuti a Gerhard Gentzen. Un altro importante formalismo per la Logica Proposizionale è il sistema detto Risoluzione, contenente una unica regola e adatto allo sviluppo di algoritmi per la soddisfacibilità e allo studio astratto della lunghezza (o complessità) delle dimostrazioni proposizionali (il campo che si occupa di questi problemi è la *Proof Complexity*).

La definizione di $T \vdash A$ include quella precedente, leggendo $\vdash A$ come $\emptyset \vdash A$. Osserviamo che questa nozione estesa di derivazione formale preserva la conseguenza logica.

Proposizione 1.7 (Correttezza)**.**

$$T \vdash B \Longrightarrow T \models B.$$

Dimostrazione. Semplice esercizio. Gli assiomi logici sono verità logiche e il Modus Ponens preserva la conseguenza logica. □

In termini insiemistici, se scriviamo $Teor(T)$ per $\{A\,;\, T \vdash A\}$ la Correttezza si esprime con la seguente inclusione:

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

ossia: il nostro calcolo permette di derivare formalmente dalle ipotesi in T *soltanto* conseguenze logiche di T.

2. Completezza Proposizionale

Il Teorema di Completezza stabilisce che

$$T \models A \text{ se e soltanto se } T \vdash A$$

Si tratta di uno dei risultati fondamentali della Logica Matematica, che sancisce l'equivalenza tra la nozione semantica di conseguenza logica $T \models A$ e la nozione sintattica di derivabilità formale $T \vdash A$.

Nel caso particolare in cui T è vuota il Teorema stabilisce la seguente equivalenza:

$$\models A \text{ se e soltanto se } \vdash A,$$

ossia A è una tautologia se e soltanto se A è un teorema del Calcolo Proposizionale.

Abbiamo già dimostrato l'implicazione: Se $T \vdash A$ allora $T \models A$; si tratta della Correttezza della nozione di derivabilità formale.

Resta da dimostare l'implicazione: Se $T \models A$ allora $T \vdash A$. In termini insiemistici si tratta della seguente inclusione:

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

che esprime il fatto che il nostro calcolo deduttivo formale permette di dedurre da una teoria *tutte* le conseguenze logiche di una teoria (questo motiva il nome di Completezza). Si noti che si tratta di una implicazione tutt'altro che banale: l'ipotesi è che tutti gli assegnamenti che soddisfano T soddisfano A mentre la conclusione è che esiste un oggetto finito che ha le proprietà richieste per essere una derivazione formale di A da premesse in T. Il Teorema di Completezza, oltre a sancire l'equivalenza di una nozione semantica ($\models$) con una nozione sintattica ($\vdash$) sancisce anche l'equivalenza di una nozione *universale infinitaria* (A è conseguenza della teoria T) con una nozione *esistenziale finitaria* (A ammette una dimostrazione con premesse in T).

Strategia per dimostrare la Completezza: $T \models A$ implica $T \vdash A$

La nostra strategia è la seguente. Dal Teorema di Compattezza sappiamo che $T \models A$ se e solo se esistono $A_1, \ldots, A_n \in T$ tali che $A_1, \ldots, A_n \models A$. Questo è equivalente a dire che la singola proposizione $(A_1 \to (A_2 \to \ldots (A_n \to A) \ldots)$ è una tautologia.

Dimostreremo come primo passo che tutte le tautologie sono teoremi del calcolo proposizionale, ossia che

$$\models B \Rightarrow \vdash B.$$

Se B è una formula del tipo $(A_1 \to (A_2 \to \ldots (A_n \to A) \ldots)$ dove $A_1, \ldots, A_n$ sono in T, abbiamo ottenuto che se $T \models A$ allora esistono $A_1, \ldots, A_n \in T$ tali che $\vdash (A_1 \to (A_2 \to \ldots (A_n \to A) \ldots)$. Da questo vogliamo ottenere $T \vdash A$. Basterà verificare che $\vdash (A_1 \to (A_2 \to \ldots (A_n \to A) \ldots)$ vale se e soltanto se $A_1, \ldots, A_n \vdash A$. A fortiori allora vale $T \vdash A$.

Esercizio: Dimostrare che $\vdash (A_1 \to (A_2 \to \ldots (A_n \to A) \ldots)$ implica $A_1, \ldots, A_n \vdash A$ e viceversa.

Analogamente $A_1, \ldots, A_n \vdash A$ se e soltanto se $\vdash (A_1 \wedge \cdots \wedge A_n) \to A$.

LOGICA MATEMATICA
ESERICIZI DI LOGICA PROPOSIZIONALE

ANNO ACCADEMICO 2023/2024

1. Esercizi di routine

Esercizio 1.1. I seguenti punti sono veri o falsi? In entrambi i casi dimostrare.

(1) $A, B \models C$ se e solo se $A \models (B \to C)$
(2) Se $A \models B$ o $A \models C$ allora $A \models (B \vee C)$
(3) Se $A \models B$ o $A \models C$ allora $(A \vee B) \models C$
(4) Se $A \models B$ o $A \models C$ allora $A \models (B \wedge C)$
(5) Se $A \models B$ e $A \models C$ allora $A \models (B \vee C)$
(6) Se $A \models B$ e $A \models C$ allora $B \models C$
(7) Se $A \models \neg A$ allora $\neg A \in$ Taut
(8) Se $A, B \models C$ allora ($A \models C$ o $B \models C$)
(9) Se $A \models C$ allora $A, B \models C$
(10) $A \models B$ se e solo se $\neg B \models \neg A$
(11) Se $A \models (B \vee C)$ allora $A \models B$ o $A \models C$
(12) $A, (A \to B), \neg(B \to C) \models D$ equivale e $A \wedge B \wedge \neg C \wedge \neg D \in$ UnSat
(13) Se esiste B tale che $A \not\models B$ allora A non è insoddisfacibile.

Esercizio 1.2. I seguenti punti sono veri o falsi? In entrambi i casi dimostrare.

(1) Se $T \models (A \vee B)$ allora $T \models A$ o $T \models B$.
(2) Sia T tale che $T \models (A \to B)$. Se $T \cup \{B\}$ è insoddisfacibile allora $T \cup \{A\}$ è insoddisfacibile.
(3) Sia T tale che $T \models (A \to B)$. Se $T \cup \{A\}$ è insoddisfacibile allora $T \cup \{B\}$ è insoddisfacibile.
(4) T è completa se e solo se per ogni A, B: $T \not\models (A \to B)$ vale se e solo se $T \models A$ e $T \models \neg B$.

Esercizio 1.3. Due proposizioni A e B si dicono *logicamente equivalenti* se per ogni assegnamento α, $\alpha(A) = \alpha(B)$. In tal caso scriviamo $A \equiv B$.

(1) associatività
   (a) $(A \vee (B \vee C)) \equiv A \vee (B \vee C)$
   (b) $(A \wedge (B \wedge C)) \equiv A \wedge (B \wedge C)$
(2) Commutatività
   (a) $(A \vee B) \equiv (B \vee A)$
   (b) $(A \wedge B) \equiv (B \wedge A)$
(3) distributività
   (a) $(A \vee (B \wedge C)) \equiv (A \vee B) \wedge (A \vee C)$
   (b) $(A \wedge (B \vee C)) \equiv (A \wedge B) \vee (A \wedge C)$
(4) Leggi di De Morgan
   (a) $\neg(A \vee B) \equiv (\neg A \wedge \neg B)$
   (b) $\neg(A \wedge B) \equiv (\neg A \vee \neg B)$
(5) Doppia Negazione $\neg\neg A \equiv A$
(6) Idempotenza
   (a) $(A \vee A) \equiv A$
   (b) $(A \wedge A) \equiv A$
(7) Interdefinibilità dei connettivi:

Note preparate da Lorenzo Carlucci, lorenzo.carlucci@uniroma1.it. Alcuni degli esercizi proposti sono tratti dai manuali di E. Mendelson, *Introduction to Mathematical Logic*, e di D. Van Dalen, *Logic and Structure*.

(a) $(A \leftrightarrow B) \equiv ((A \to B) \wedge (B \to A))$
(b) $(A \to B) \equiv (\neg A \vee B)$
(c) $(A \vee B) \equiv (\neg A \to B)$
(d) $(A \vee B) \equiv \neg(\neg A \wedge \neg B)$
(e) $(A \wedge B) \equiv \neg(\neg A \vee \neg B)$

Esercizio 1.4. Se conveniamo di usare — all'interno di proposizioni — la costante 1 (o il simbolo $\top$) al posto di una qualunque tautologia e la costante 0 (o il simbolo $\bot$) al posto di una qualunque formula insoddisfacibile, allora valgono le seguenti leggi algebriche aggiuntive.

(1) Assorbimento
   (a) $(A \vee 0) \equiv A$
   (b) $(A \wedge 1) \equiv A$
(2) Contraddizione, Terzo Escluso
   (a) $(A \vee \neg A) \equiv 1$
   (b) $(A \wedge \neg A) \equiv 0$

Esercizio 1.5. La relazione di equivalenza logica (definita come $A \equiv B$ se e solo se $\models (A \leftrightarrow B)$) è invece una relazione di equivalenza sull'insieme delle proposizioni.

(1) $A \equiv A$
(2) Se $A \equiv B$ e $B \equiv C$ allora $A \equiv C$
(3) Se $A \equiv B$ allora $B \equiv A$.

Esercizio 1.6. Se so che $A \to B$ ha valore 1, che cosa posso concludere del valore di verità delle proposizioni seguenti?

$$((A \vee C) \to (B \vee C)), \quad ((A \wedge C) \to (B \wedge C)), \quad ((\neg A \wedge B) \leftrightarrow (A \vee B))$$

Esercizio 1.7. Se so che $A \leftrightarrow B$ ha valore 0, che cosa posso concludere del valore di verità delle proposizioni seguenti?

$$(A \wedge B), \quad (A \vee B), \quad (A \to B), \quad ((A \wedge C) \leftrightarrow (B \wedge C))$$

Esercizio 1.8. Se so che $A \leftrightarrow B$ ha valore 1, che cosa posso concludere del valore di verità delle proposizioni seguenti?

$$(A \wedge B), \quad (A \vee B), \quad (A \to B), \quad ((A \wedge C) \leftrightarrow (B \wedge C))$$

Esercizio 1.9. Dimostrare che

$$A \wedge (A \vee B) \equiv A$$

(Suggerimento: dimostrare che $A \to (A \wedge (A \vee B))$ e $(A \wedge (A \vee B)) \to A$ sono entrambe tautologie)

Esercizio 1.10. Dimostrare le seguenti equivalenze logiche.

(1) $A \to B \equiv A \leftrightarrow (A \wedge B)$
(2) $A \to B \equiv B \leftrightarrow (A \vee B)$
(3) $A \wedge B \equiv (A \leftrightarrow B) \leftrightarrow (A \vee B)$
(4) $A \leftrightarrow B \equiv (A \vee B) \to (A \wedge B)$

Esercizio 1.11. Formalizzare i seguenti enunciati usando il linguaggio proposizionale composto da variabili $f_{i,j}$ per $i \in \{1,2,3\}$ e $j \in \{1,2,3,4\}$ con il significato intuitivo di $f(i) = j$.

(1) f è la funzione costante con dominio $\{1,2,3\}$ e valore 4.
(2) f è una relazione funzionale (i.e., una funzione) con dominio $\{1,2,3\}$ e immagine contenuta in $\{1,2,3,4\}$.
(3) f è una funzione suriettiva con dominio contenuto in $\{1,2,3\}$ e codominio $\{1,2,3,4\}$.
(4) f è una funzione iniettiva con dominio $\{1,2,3\}$ e codominio $\{1,2,3,4\}$.

Esercizio 1.12. Formalizzare i seguenti enunciati usando il linguaggio proposizionale composto da variabili a_i e b_i con $i \in \{1,2,3,4\}$ con significato intuitivo $i \in A$ e $i \in B$, rispettivamente.

(1) A è un sottinsieme non vuoto di $\{1,2,3,4\}$.
(2) A e B sono sottinsiemi non vuoti di $\{1,2,3,4\}$ tali che $A \cap B = \varnothing$.
(3) A e B sono sottinsiemi non vuoti di $\{1,2,3,4\}$ tali che $A \cup B = \{1,2,3,4\}$.

(4) A e B sono sottinsiemi non vuoti di $\{1,2,3,4\}$ tali che $A \subset B$ e $A \neq B$.

Esercizio 1.13. Individuare un linguaggio proposizionale $\mathcal{L}$ e una teoria T in $\mathcal{L}$ che catturi la nozione di "essere una configurazione lecita nel gioco del Tris".

Esercizio 1.14. Individuare un linguaggio proposizionale $\mathcal{L}$ e una teoria T in $\mathcal{L}$ che catturi la nozione di "essere una soluzione a un Sudoku 9×9".

Esercizio 1.15. Vero o Falso?

(1) $((p_1 \to p_2) \wedge (\neg p_3 \to \neg p_2) \wedge (p_1 \wedge \neg p_3)) \in Unsat$;
(2) Se $\neg A$ è una tautologia allora $A \vee (A \to C)$ è una tautologia;
(3) Se A non è insoddisfacibile allora esiste un B soddisfacibile tale che $B \models A$ e $A \not\models B$.

Esercizio 1.16. Se $A \notin UnSat$ allora esiste $B \notin UnSat$ tale che $B \models A$ e $A \not\models B$.

Esercizio 1.17. Trovare A, B tali che $\neg(A \wedge B) \in Taut$, trovare A, B tali che $\neg(A \to B) \in Taut$, trovare A, B tali che $\neg(A \wedge B)$ e $\neg(A \to B)$ siano entrambe in $Taut$.

Esercizio 1.18. Consideriamo la seguente formula

$$p\#(q \to (p * q))$$

Per quale scelta di connettivi da sostituire a $\#$ e $*$ (nell'ordine) la proposizione risultante è una tautologia?

Esercizio 1.19. Se A contiene n connettivi, allora ha contiene $\leq 2n+1$ sottoformule.

Esercizio 1.20. Definire per ricorsione una funzione $S : \mathrm{PROP} \to \mathcal{P}(PROP)$ tale che $S(A) = \{B : B \text{ è sottoformula di } A\}$.

Esercizio 1.21. Dimostrare che per ogni assegnamento α

(1) $\alpha(A \wedge B) = \alpha(A) \cdot \alpha(B)$,
(2) $\alpha(A \vee B) = \alpha(A) + \alpha(B) - \alpha(A) \cdot \alpha(B)$,
(3) $\alpha(A \to B) = 1 - \alpha(A) + \alpha(A) \cdot \alpha(B)$,
(4) $\alpha(A \leftrightarrow B) = 1 - |\alpha(A) - \alpha(B)|$.

Esercizio 1.22. Dimostrare che se $\models (A \to B)$ allora $\models (A \wedge B) \leftrightarrow A$ e $\models (A \vee B) \leftrightarrow B$.

Esercizio 1.23. Definiamo una mappa $^+$ da proposizioni a proposizioni come segue.

- $A^+ = \neg A$ se A è una variabile proposizionale,
- $(A \wedge B)^+ = A^+ \vee B^+$,
- $(A \vee B)^+ = A^+ \wedge B^+$,
- $(\neg A)^+ = \neg A^+$.

Dimostrare che $\models \neg A \leftrightarrow A^+$.

2. Teoremi Generali

2.1. **Teoremi di Sostituzione.** I seguenti tre punti sono ragionevoli:

(1) Se A è una tautologia, se sostituisco in A una variabile proposizionale con una formula qualunque, ottengo ancora una tautologia.
(2) Se A e B sono equivalenti, e sostituisco sia in A che in B una stessa variabile proposizionale con una stessa formula qualunque, ottengo due formule equivalenti.
(3) Se sostituisco in una stessa formula A una variabile proposizionale con due formule equivalenti, ottengo due formule equivalenti.

Per dimostrarli introduciamo un po' di notazione. Siano $A, B_1, \ldots, B_n$ proposizioni, siano $p_1, \ldots, p_n$ variabili proposizionali distinte. Denotiamo con $A[p_1/B_1 \ldots p_n/B_n]$ il risultato di sostituire simultaneamente nella proposizione A la variabile proposizionale p_i con la formula B_i, per ogni $i \in \{1, \ldots, n\}$. Nota bene: la sostituzione è un'operazione puramente sintattica che trasforma proposizioni in proposizioni, e la sostituzione deve essere simultanea, non sequenziale! La proposizione $A[p_1/B_1 \ldots p_n/B_n]$ può essere definita rigorosamente per ricorsione (Esercizio!).

Esercizio 2.1. Dimostrare il seguente Teorema.

Teorema 2.2. *Siano $A, B_1, \dots, B_n$ proposizioni, siano $p_1, \dots, p_n$ variabili proposizionali distinte. Abbreviamo $A[p_1/B_1 \dots p_n/B_n]$ con A^*. Sia α un assegnamento. Definiamo un nuovo assegnamento $\alpha^* : \mathcal{L} \to \{0,1\}$ (in funzione di α, p_i, B_i) come segue.*

$$\alpha^*(Q) = \begin{cases} \alpha(Q) & \text{se } Q \neq P_i \text{ per ogni } i \in \{1, \dots, n\} \\ \alpha(B_i) & \text{se } Q = P_i \text{ per qualche } i \in \{1, \dots, n\}. \end{cases}$$

Allora

$$\alpha(A^*) = \alpha^*(A).$$

Esercizio 2.3. Dimostrare la seguente proposizione.

Proposizione 2.4 (Sostituzione in tautologie)**.** *Siano $A, B_1, \dots, B_n$ proposizioni, siano $p_1, \dots, p_n$ variabili proposizionali distinte. Se A è una tautologia allora $A[p_1/B_1 \dots p_n/B_n]$ è una tautologia.*

Esercizio 2.5. Dimostrare la seguente proposizione.

Proposizione 2.6 (Sostituzione in formule equivalenti)**.** *Siano $C, D, B_1, \dots, B_n$ proposizioni, siano $p_1, \dots, p_n$ variabili proposizionali distinte. Se $\models (C \leftrightarrow D)$ allora*

$$\models (C[p_1/B_1 \dots p_n/B_n]) \leftrightarrow (D[p_1/B_1 \dots p_n/B_n]).$$

Esercizio 2.7. Dimostrare il Lemma seguente.

Lemma 2.8. *Per ogni assegnamento α e formula A, B: $\models (A \to B)$ se e solo se $\alpha(A) \leq \alpha(B)$.*

Esercizio 2.9. Dimostrare il Teorema seguente.

Teorema 2.10. *Siano A, B_1, B_2 proposizioni e sia p una variabile proposizionale. Allora per ogni assegnamento α vale*

$$\alpha(B_1 \leftrightarrow B_2) \leq \alpha(A[p/B_1] \leftrightarrow A[p/B_2]).$$

2.2. **Dualità.**

Definizione 2.11. Definiamo una mappa ^{d} da proposizioni in proposizioni.

$$A^d = A \text{ se } A \text{ è una variabile.}$$
$$(B \wedge C)^d = (B^d \vee C^d)$$
$$(B \vee C)^d = (B^d \wedge C^d)$$
$$(\neg B)^d = \neg B^d$$

Vale il seguente Teorema di Dualità.

Teorema 2.12 (Teorema di Dualità)**.** *$A \equiv B$ se e solo se $A^d \equiv B^d$.*

Esercizio 2.13. Dimostrare il Teorema di Dualità e commentarne il significato.

2.3. **Forme Normali.** Chiamiamo "letterale" una variabile proposizionale o una negazione di una variabile proposizionale. Diciamo che A è in Forma Normale Congiuntiva (CNF) se A è una congiunzione di disgiunzioni di letterali, ossia è della forma seguente, dove gli $A_{i,j}$ sono letterali.

$$(A_{1,1} \vee A_{1,2} \vee \dots \vee A_{1,m_1}) \wedge (A_{2,1} \vee A_{2,2} \vee \dots \vee A_{2,m_2}) \dots \wedge (A_{n,1} \vee A_{n,2} \vee \dots \vee A_{n,m_n})$$

Diciamo che A è in Forma Normale Disgiuntiva (DNF) se A è una disgiunzione di congiunzioni di letterali, ossia è della forma seguente, dove gli $A_{i,j}$ sono letterali.

$$(A_{1,1} \wedge A_{1,2} \wedge \dots \wedge A_{1,m_1}) \vee (A_{2,1} \wedge A_{2,2} \wedge \dots \wedge A_{2,m_2}) \dots \vee (A_{n,1} \wedge A_{n,2} \wedge \dots \wedge A_{n,m_n}).$$

Usiamo $\bigwedge_{i \leq n} A_i$ come abbreviazione di

$$A_1 \wedge A_2 \wedge \dots \wedge A_n.$$

e analogamente $\bigvee_{i \leq n} A_i$ come abbreviazione di

$$A_1 \vee A_2 \vee \dots \vee A_n.$$

Conquesta notazione, A è una CNF se è della forma

$$\bigwedge_{i\leq n}\bigvee_{j\leq m_i} A_{i,j},$$

ed è in DNF se è della forma

$$\bigvee_{i\leq n}\bigwedge_{j\leq m_i} A_{i,j},$$

dove gli $A_{i,j}$ sono letterali.

Vale il seguente Teorema di Forma Normale.

Teorema 2.14 (Forme Normali Congiuntive e Disgiuntive)**.** *Per ogni A esiste A^{CNF} e A^{DNF} tali che A^{CNF} è una CNF, A^{DNF} è una DNF, e*

$$\models A \leftrightarrow A^{\mathrm{CNF}},$$

$$\models A \leftrightarrow A^{\mathrm{DNF}},$$

Esercizio 2.15. Dimostrare il Teorema di Forma Normale. Suggerimento: una dimostrazione si ottiene descrivendo la tavola di verità di A. Anche una dimostrazione induttiva è a portata di mano.

2.4. **Algebre Booleane.** Fissiamo un linguaggio proposizionale $\mathcal{L}$. Consideriamo la relazione $\leq$ su formule di $\mathcal{L}$:

$$A \leq B \text{ se e solo se } \models A \to B.$$

Esercizio 2.16. Dimostrare che $\leq$ è simmetrica e transitiva.

Esercizio 2.17. Determinare se $\leq$ è antisimmetrica.

Definiamo $\equiv$ sull'insieme delle proposizioni come segue:

$$A \equiv B \text{ se e soltanto se } (\models A \to B \text{ e } \models B \to A).$$

Esercizio 2.18. Dimostrare che $\equiv$ è una relazione di equivalenza.

Esercizio 2.19. Dimostrare che se $A \equiv A'$ e $B \equiv B'$ allora $\models A \to B$ se e solo se $\models A' \to B'$.

Se A è una formula indichiamo con $[A]_{\equiv}$ la sua classe di equivalenza relativamente alla relazione $\equiv$ sopra definita.

Definiamo la relazione $\leq$ sul quoziente $\mathrm{PROP}/\equiv$ come segue:

$$[A]_{\equiv} \leq [B]_{\equiv} \text{ se e soltanto se } \models A \to B.$$

Esercizio 2.20. Dimostrare che la definizione di $\leq$ è ben posta.

Si definisce *Algebra di Boole* un reticolo complementato distributivo con almeno due elementi.

Si ricorda che un *reticolo* è un ordine parziale $(X, \leq)$ in cui ogni sottinsieme di due elementi $\{x, x'\}$ ha sia un infimo che un supremo. Si ricorda che un *supremo* di un sottinsieme A di un ordine parziale $(X, \leq)$ è un elemento s che è limite superiore di A e minimale: ogni elemento di A è $\leq s$ e per ogni altro limite superiore t di A vale $s \leq t$. Un *infimo* è definito in maniera duale come limite inferiore di A massimale. Si denota con $x \wedge y$ l'infimo di $\{x, y\}$ e con $x \vee y$ il supremo di $\{x, y\}$.

Un reticolo $(X, \leq)$ è *complementato* se possiede un minimo (denotato 0), un massimo (denotato 1) e per ogni $x \in X$ esiste $y \in X$ tale che $x \vee y = 1$ e $x \wedge y = 0$. Un tale y viene detto il *complemento* di x.

Un reticolo $(X, \leq)$ è *distributivo* se per ogni $x, y, z \in X$ valgono

$$(x \vee y) \wedge z = (x \vee z) \wedge (y \vee z)$$

, e

$$(x \wedge y) \vee z = (x \wedge z) \vee (x \wedge z).$$

Esercizio 2.21. Dimostrare che $(\mathrm{PROP}/\equiv, \leq)$ è un'Algebra di Boole.

2.5. **Interpolazione.** Estendiamo per convenienza la nozione di formula introducendo due costanti proposizionali $\top$ e $\bot$ ponendo che $\top$ è vera sotto ogni assegnamento e $\bot$ è falsa sotto ogni assegnamento. Le costanti $\top, \bot$ si comportano come variabili proposizionali: $\top$ e $\bot$ sono formule, e così $\top \wedge A$, $\bot \to B$, $\neg\top$, etc.

Esercizio 2.22. Se p_1 non compare in B e p_2 non compare in A allora:

(1) Se $\models A \to B$ allora $\models A[p_1/C] \to B$ per ogni formula C.
(2) Se $\models A \to B[p_2]$ allora $\models A \to B[p_2/C]$ per ogni formula C.

Esercizio 2.23. Sia A una formula contenente la variabile p_1. Dimostrare i seguenti punti:

(1) $A[p_1/\top] \models A[p_1/\top] \leftrightarrow \top$
(2) $A[p_1/\top] \models A[p_1/A[p_1/\top]]$
(3) $\neg A[p_1/\top] \models A[p_1/\top] \leftrightarrow \bot$
(4) $A, \neg A[p_1/\top] \models p_1 \leftrightarrow \bot$
(5) $A, \neg A[p_1/\top] \models A[p_1/A[p_1/\top]]$
(6) $A \models A[p_1/A[p_1/\top]]$

Siano A, B tali che $\models A \to B$. Una proposizione C è un *interpolante* di A e B se C contiene solo variabili comuni a A e B e valgono:

$$\models A \to C; \text{ e } \models C \to B.$$

Esercizio 2.24. Siano A, B tali che $\models A \to B$. Supponiamo che A contenga esattamente le variabili p_1, p_2 e B le variabili p_2, p_3. Dimostrare che $A[p_1/A[p_1/\top]]$ è un interpolante di A e B.

Vale il seguente Teorema di Interpolazione.

Teorema 2.25 (Teorema di Interpolazione)**.** *Per ogni A, B tali che $\models A \to B$ esiste un interpolante.*

Esercizio 2.26. Dimostrare il Teorema di Interpolazione.

2.6. **Teorie indipendenti.** Una teoria T è detta *indipendente* se per ogni $A \in T$ vale $A \setminus \{A\} \not\models A$.

Esercizio 2.27. Se T è finita allora esiste $T_0 \subseteq T$ indipendente tale che per ogni $A \in T$ vale $T_0 \models A$.

Esercizio 2.28. Se T è numerabile ($T = \{A_1, A_2, \dots\}$) esiste una teoria $T' = \{B_1, B_2, \dots\}$ tale che per ogni n:

$$T \models B_n \text{ e } T' \models A_n,$$

e tale che

$$\models B_{n+1} \to B_n,$$

ma

$$\not\models B_n \to B_{n+1}.$$

Esercizio 2.29. Sia T' come nell'esercizio precedente e supponiamo che sia numerabile. Poniamo

$$C_1 = B_1, C_{n+1} = B_n \to B_{n+1}.$$

Dimostrare che $T^* = \{C_1, C_2, \dots\}$ è indipendente e tale che per ogni n:

$$T^* \models B_n \text{ e } T' \models C_n.$$

Dedurre che per ogni teoria T esiste una teoria T^* indipendente ed equivalente a T nel senso che per ogni $A \in T$ si ha $T^* \models A$ e viceversa per ogni $A \in T^*$ si ha $T \models A$. T^* deve essere necessariamente un sottinsieme di T?

Esercizio 2.30. Sia $\mathcal{L}$ il linguaggio $\{p_1, p_2, \dots\}$. Dimostrare che la teoria $T = \mathcal{L}$ è completa.

3. Problemi

Esercizio 3.1. Sia T una teoria infinita numerabile, $T = \{A_1, A_2, \dots, \}$. Supponiamo che per ogni assegnamento α esista un n tale che $\alpha(A_n) = 1$. Allora esiste un m tale che $\models A_1 \vee \cdots \vee A_m$.

Esercizio 3.2. Supponiamo di introdurre nel nostro formalismo disgiunzioni e congiunzioni infinite. Il Teorema di Compattezza continua a valere?

Esercizio 3.3. Dato un assegnamento α per il linguaggio $\{p_{i,j} : i, j \in \mathbf{N}\}$ consideriamo $G_\alpha = (V_\alpha, E_\alpha)$ con $V_\alpha = \mathbf{N}$ e $E_\alpha = \{(i, j) : \alpha(p_{i,j}) = 1\}$. Dimostrare che non esiste una teoria T tale che $\alpha(T) = 1$ se e solo se G_α è un grafo connesso su $\mathbf{N}$.

Esercizio 3.4. Come sopra sostituendo $\mathbf{N}$ con $\mathbf{R}$.

3.1. **Principio dei Piccioni Schizzinosi.** Consideriamo la seguente variante del Principio della Piccionaia. Abbiamo un insieme A di piccioni e un insieme B di piccionaie monoposto. Ogni piccione in A ha un insieme di piccioniaie preferite B: tollera di accomodarsi solo ed esclusivamente in una delle sue piccionaie preferite.

A che condizioni è possibile soddisfare tutti i piccioni in A allocando ciascuno di essi in una delle sue piccionaie preferite senza conflitti?

Se A è finito, è ovvio osservare che l'allocazione è impossibile se l'insieme delle piccionaie preferite da qualche piccione in A ha cardinalità minore di A. Analogamente se per qualche $k \leq |A|$ esiste un insieme di k piccioni che determina un insieme di meno di k piccionaie preferite, è impossibile allocare i k piccioni schizzinosi senza conflitti. Questa condizione risulta anche sufficiente. Vale il seguente Lemma.

Lemma 3.5. *Se A è un insieme di m piccioni e per ogni $k \leq m$ ogni insieme di k piccioni in A determina almeno k piccionaie preferite, allora è possibile accomodare ogni piccione di A in una piccionaia di suo gusto senza creare conflitti.*

Esercizio 3.6. Dimostrare il Lemma per induzione.

Consideriamo ora lo stesso problema ma per insiemi anche infiniti. Abbiamo quindi un insieme A di piccioni schizzinosi e un insieme B di piccionaie. Ogni piccione in A seleziona un certo numero di piccionaie preferite in B. Se ammettiamo che ogni piccione ha un numero finito di piccionaie preferite, vallora vale il seguente Teorema.

Teorema 3.7. *Sia A un insieme di piccioni e B un insieme di piccionaie, tali che ogni piccione in A ha un numero finito di piccionaie preferite in B. Se per ogni $k \in \mathbf{N}$ ogni insieme di k piccioni in A determina almeno k piccionaie preferite allora è possibile accomodare ogni piccione in A in una delle sue piccionaie preferite in B senza creare conflitti.*

Esercizio 3.8. Dimostrare il Teorema per Compattezza assumendo il Lemma.

Esercizio 3.9. Il Teorema vale se si ammette che qualche piccione ha infinite piccionaie preferite? (NB: questo non è un esercizio di Logica).

3.2. **Catene in Ordini Parziali.** Due elementi x, y in un ordine parziale $(X, \leq)$ sono *confrontabili* se vale $x \leq y$ o $y \leq x$, e *inconfrontabili* altrimenti. Una *catena* in X è un sottinsieme di X costituito da elementi due a due confrontabili.

Esercizio 3.10. Dimostrare che se ogni sottinsieme finito di X è una unione di k catene allora X è una unione di k catene.

Un'anticatena in $(X, \leq)$ è un sottinsieme di X di elementi due a due inconfrontabili.

Teorema 3.11. *Sia $(X, \leq)$ un ordine parziale finito. $(X, \leq)$ è unione di ℓ catene se e solo se ogni anticatena ha al più ℓ elementi.*

Esercizio 3.12. Dimostrare che il Teorema precedente implica la sua versione generale (per insiemi finiti e infiniti).

Esercizio 3.13. ** Definire un calcolo formale basato sulla seguente idea: il calcolo dimostra equivalenze logiche $A \equiv B$ stabilendo catene finite di equivalenze tra formule $A_1 \equiv A_2 \equiv A_3 \equiv \cdots \equiv A_n$, dove ogni equazione deve essere giustificata in base a una delle leggi logiche notevoli e ai teoremi di sostituzione (cf. esercizi precedenti). Definire il formalismo e la nozione di deduzione formale e dimostrare un teorema di completezza e correttezza del calcolo definito: Correttezza: Se una equivalenza è derivabile, è una vera equivalenza logica? Completezza: Per ogni equivalenza logica è possibile ottenerne una deduzione formale nel formalismo definito?

3.3. **Compattezza logica e topologica.** Sia S l'insieme di tutti gli assegnamenti per un linguaggio $\mathcal{L} = \{p_1, p_2, \dots\}$. Per ogni formula A sia X_A l'insieme degli assegnamenti che la soddisfano.

Esercizio 3.14. Dimostrare che gli insiemi X_A formano una base per una topologia su S, ossia che

(1) Gli X_A coprono S; e
(2) Se $\alpha \in X_A \cap X_B$ allora esiste C tale che $\alpha \in X_C$ e $X_C \subseteq X_A \cap X_B$.

Esercizio 3.15. Dimostrare che per ogni proposizione A l'insieme X_A è chiuso (nella topologia dell'esercizio precedente). (Suggerimento: per semplicità considerare dapprima A una congiunzione contenente p_i o $\neg p_i$ per ogni $i \in [1, n]$ per qualche n. Considerare poi A in forma normale disgiuntiva.)

Esercizio 3.16. Il Teorema di Compattezza proposizionale implica la compattezza della topologia definita negli esercizi precedenti.

(Suggerimento: considerare una collezione C di insiemi chiusi con la proprietà che ogni sottinsieme finito ha intersezione non vuota. Basta dimostrare che è non vuota anche l'intersezione di tutti gli elementi della collezione. A questo scopo consideriamo la teoria T contenente la negazione di ogni proposizione A tale che X_A è contenuto nel complemento di qualche chiuso della collezione C).

Esercizio 3.17. La compattezza della topologia definita negli esercizi precedenti implica il Teorema di Compattezza proposizionale.

Esercizio 3.18. Dimostrare che il Teorema di Compattezza proposizionale è equivalente al seguente teorema topologico: Per ogni Y lo spazio topologico prodotto $\{0, 1\}^Y$ è uno spazio compatto.

4. Domande d'esame

Esercizio 4.1. Indicare se le seguenti affermazioni sono vere o false, dove A, B, C, D variano su arbitrarie proposizioni in un linguaggio proposizionale.

(1) $\neg A \vee B, B \to C, \neg D \vee A \models D \to C$
(2) Se $\neg A \models B$ allora $A \not\models B$.
(3) Se $\neg A \models B$ allora $\neg B \models A$.

Esercizio 4.2. Indicare se le seguenti affermazioni sono vere o false, dove A, B, C variano su arbitrarie proposizioni in un linguaggio proposizionale e T è una teoria proposizionale soddisfacibile.

(1) Se $T \not\models A$ allora $T \models \neg A$.
(2) Se $T, A \models B$ allora se $T \cup \{A\}$ è insoddisfacibile allora $T \models \neg B$.
(3) Se $T, A \vee B \models C$ e $T \models B$ allora $T \models A \to C$.

Esercizio 4.3. Indicare se le seguenti proposizioni sono vere o false, dove A, B, C variano su arbitrarie proposizioni in un linguaggio proposizionale e assumiamo che $A \leftrightarrow B$ abbia valore FALSO.

(1) $(A \vee B)$ è sicuramente vera.
(2) $(A \wedge C) \leftrightarrow (B \wedge C)$ è sicuramente vera.
(3) $(A \wedge \neg B) \leftrightarrow (\neg A \wedge B)$ può essere sia vera che falsa.

Esercizio 4.4. Indicare se le seguenti affermazioni sono vere o false, dove A, B, C variano su arbitrarie proposizioni in un linguaggio proposizionale.

(1) $\neg A, (\neg B \wedge C) \vee (A \wedge C) \models \neg B \wedge C$.
(2) Se $B \to B \models A$ allora $\neg A$ è insoddisfacibile.
(3) $(\neg\neg A \to \neg A) \models \neg A$.

Esercizio 4.5. Indicare se le seguenti affermazioni sono vere o false, dove A, B, C variano su arbitrarie proposizioni in un linguaggio proposizionale e T è una teoria proposizionale soddisfacibile.

(1) $T \models A$ oppure $T \models \neg A$.
(2) Se $T \models A$ allora $T \not\models \neg A$.
(3) Se $T \models (A \leftrightarrow B)$ allora se $T \cup \{A\}$ è soddisfacibile anche $T \cup \{B\}$ è soddisfacibile.
(4) Se $T \models A$ e $T \models (A \to B) \wedge C$ allora $T \models B \vee C$.
(5) Se $T, A \vee B \models C$ e $T \models B$ allora $T \models C$.
(6) Per ogni A, B vale $T \models A \to B$ oppure $T \models B \to A$.

Esercizio 4.6. Indicare se le seguenti proposizioni sono vere o false, dove A, B, C variano su arbitrarie proposizioni in un linguaggio proposizionale e assumiamo che $A \leftrightarrow B$ abbia valore FALSO.

(1) $(A \vee \neg B)$ è sicuramente vera.
(2) $(A \wedge C) \leftrightarrow (B \wedge C)$ è sicuramente vera.
(3) $(A \wedge \neg B) \to (\neg A \wedge B)$ può essere sia vera che falsa.

Esercizio 4.7. Indicare se le seguenti affermazioni sono vere o false, dove A, B, C variano su arbitrarie proposizioni in un linguaggio proposizionale.

(1) $A, B \models C$ se e solo se $A \models (B \to C)$.
(2) Se $A \models \neg A$ allora $\neg A$ è una tautologia.
(3) Se $A \models B$ allora $\neg B \models \neg A$.

Esercizio 4.8. Indicare se le seguenti affermazioni sono vere o false, dove A, B, C variano su arbitrarie proposizioni in un linguaggio proposizionale e T è una teoria proposizionale soddisfacibile.

(1) Se $B \not\models A$ allora $B \models \neg A$.
(2) Se $T \models A \vee B$ allora $T \not\models A \wedge \neg B$.
(3) Se $T, A \models \neg B$ allora se $T \cup \{\neg A\}$ è insoddisfacibile allora $T \models \neg B$.
(4) Se $T \models A$ e $T \models (\neg A \vee B) \wedge C$ allora $T, \neg B \models C$.
(5) Se $T, \neg A \to B \models C$ e $T \models B$ allora $T \models A \to C$.
(6) Per ogni A, B vale $T \models \neg A \vee B$ oppure $T \models A \vee \neg B$.

Esercizio 4.9. Sia $\mathcal{L}$ il linguaggio proposizonale contenente variabili $p_{i,j}$ con $i, j \in \mathbf{N}$. Interpretando $p_{i,j}$ come "l'immagine di i via f è j" scrivere una proposizione o un insieme di proposizioni che esprimano il fatto che una relazione $f \subseteq \{1,2,3,4,5\} \times \{1,2,3,4\}$ è una funzione ed èiniettiva.

Esercizio 4.10. Indicare se le seguenti affermazioni sono vere o false, dove A, B, C variano su arbitrarie proposizioni in un linguaggio proposizionale.

(1) $A, B \models C$ se e solo se $A \models (B \to C)$.
(2) Se $A \models B$ o $A \models C$ allora $A \models (B \vee C)$.
(3) Se $A \models \neg A$ allora $\neg A$ è una tautologia.
(4) Se $A \models C$ allora $A, B \models C$.
(5) Se $A \models B$ allora $\neg B \models \neg A$.
(6) Se $A \models (B \vee C)$ allora, se $A \not\models B$ allora $A \models C$.

Esercizio 4.11. Indicare se le seguenti affermazioni sono vere o false, dove A, B, C variano su arbitrarie proposizioni in un linguaggio proposizionale e T è una teoria proposizionale nello stesso linguaggio.

(1) Se $T \models (A \vee B)$ allora $T \models A$ o $T \models B$.
(2) Se $T \models (A \to B)$ allora se $T \cup \{B\}$ è insoddisfacibile $T \cup \{A\}$ è insoddisfacibile.
(3) Se $T \models (A \to B)$ allora se $T \cup \{A\}$ è insoddisfacibile $T \cup \{B\}$ è insoddisfacibile.
(4) Se A è una tautologia allora $A \vee (A \vee C)$ è una tautologia.
(5) Se $A \vee (A \vee C)$ è una tautologia allora A è una tautologia.
(6) Per ogni A, B vale $A \models B$ o $B \models A$.

Esercizio 4.12. Indicare se le seguenti affermazioni sono vere o false, dove A, B, C variano su arbitrarie proposizioni in un linguaggio proposizionale.

(1) Se $A \vdash (B \to C)$ se e solo se $B \vdash (A \to C)$.

(2) $A \vdash (B \to C)$ se e solo se $B, \neg C \vdash A$.
(3) $(\neg B \to \neg A) \vdash (A \to B)$.

Esercizio 4.13. Indicare se le seguenti affermazioni sono vere o false, dove A, B, C variano su arbitrarie proposizioni in un linguaggio proposizionale.

(1) $\neg A, (B \to A) \wedge C \models \neg B \wedge C$.
(2) Se $B \vee \neg B \models A$ allora A è una tautologia.
(3) $(A \to \neg A) \models \neg A$.
(4) Se $A \models \neg B$ allora $A \models \neg(B \to C)$.
(5) Se $A \models B$ allora $\neg B \models (A \to C)$.
(6) Se $(B \wedge C) \not\models (A \vee \neg C)$ allora $\neg A \wedge B \wedge C$ è soddisfacibile.

Esercizio 4.14. Indicare se le seguenti affermazioni sono vere o false, dove A, B, C variano su arbitrarie proposizioni in un linguaggio proposizionale e T e S sono teorie proposizionali nello stesso linguaggio.

(1) Se $T \models A$ e $S \models \neg A$ allora $T \cap S \models A \wedge \neg A$.
(2) Se A è soddisfacibile sse B è soddisfacibile, allora $\models (A \leftrightarrow B)$.
(3) Se $T \models (A \leftrightarrow B)$ allora se $T \cup \{A\}$ è insoddisfacibile anche $T \cup \{B\}$ è insoddisfacibile.
(4) Se $T \models A$ e $S \models A \to B$ allora $T \cup S \models B$.
(5) Se $T, A \vee B \models C$ e $S \models B$ allora $T \cup S \models A$.
(6) Per ogni A, B vale $\models A \to B$ oppure $\models B \to A$.

Esercizio 4.15. Indicare se le seguenti proposizioni sono vere o false, dove A, B, C variano su arbitrarie proposizioni in un linguaggio proposizionale e assumiamo che $A \leftrightarrow B$ abbia valore FALSO.

(1) $(A \vee B)$ è sicuramente vera.
(2) $(A \wedge C) \leftrightarrow (B \wedge C)$ è sicuramente vera.
(3) $(A \wedge \neg B) \leftrightarrow (\neg A \wedge B)$ può essere sia vera che falsa.

LOGICA MATEMATICA

A.A. 23/24, LOGICA PREDICATIVA, DISPENSA N. 1

SOMMARIO. Discutiamo le limitazioni espressive della Logica Proposizionale. Motiviamo informalmente il linguaggio della Logica dei Predicati.

1. Limitazioni espressive della Logica Proposizionale

In questa sezione motiviamo la necessità di estendere la logica proposizionale a una logica più espressiva, la logica dei predicati.

Procediamo così: analizziamo alcuni esempi abitualmente usati per illustrare le limitazioni della logica proposizionale. Ne diamo formalizzazioni proposizionali e analizziamo in quali casi e sotto quali aspetti queste formalizzazioni sono carenti. Vediamo in che senso la logica predicativa supplisce a queste carenze. Motiviamo attraverso esempi i costrutti fondamentali della logica predicativa: costanti, variabili, quantificatori, predicati, relazioni.

Esempio 1.1. Il seguente è un classico esempio che viene presentato per illustrare le limitazioni della logica proposizionale (vedi e.g., E. Mendelson, *Introduzione alla Logica Matematica*).

> Ogni amico di Marco è amico di Pietro.
> Claudio non è amico di Pietro.
> Dunque Claudio non è amico di Marco.

Proponiamo la seguente formalizzazione in logica proposizionale. È naturale assumere che il dominio di discorso di interesse per questo esempio, ossia l'insieme degli esseri umani, è un dominio finito. Sia U il numero degli essere umani. Usiamo un linguaggio composto da variabili $p_{i,j}$, che intuitivamente stanno per "i è amico di j", dove i, j variano su un insieme finito di indici $\{1, \dots, U\}$. Assumiamo inoltre che nella numerazione $\{1, \dots, U\}$ degli esseri umani Marco sia il numero 1, Pietro il numero 2 e Claudio il numero 3. Possiamo allora formalizzare l'argomento precedente come segue, riducendo le quantificazioni universali a congiunzioni.

$$(p_{1,1} \to p_{1,2}) \wedge (p_{2,1} \to p_{2,2}) \wedge \dots \wedge (p_{U,1} \to p_{U,2})$$

$$\neg p_{3,2}$$

$$\neg p_{3,1}$$

Si verifica facilmente che le prime due proposizioni implicano logicamente la terza.

Perché le formalizzazioni proposte non sono soddisfacenti? Perché gli esempi di sopra ci indicano che dobbiamo estendere la logica proposizionale? Consideriamo due nuove versioni degli esempi fatti.

Esempio 1.1 (Revisited)
Consideriamo la seguente variante dell'Esempio 1.

> Ogni numero intero inferiore a 5 è inferiore a 10.
> 20 non è inferiore a 10
> Dunque 20 non è inferiore a 5

La struttura dell'argomento è identica a quella dell'Esempio 1. Cosa succede se proviamo a formalizzarlo in logica proposizionale come abbiamo fatto per l'Esempio 1? La differenza fondamentale è che il dominio del discorso in questo caso consiste di un numero infinito di elementi (gli interi). Per formalizzare la prima

Note preparate da Lorenzo Carlucci, lorenzo.carlucci@uniroma1.it.

premesse (con quantificazione universale) dovremmo allora usare congiunzioni infinite, o perlomeno usare un insieme infinito di premesse, e.g.,

$$(p_{-100,5} \to p_{-100,10}), (p_{-99,5} \to p_{-99,10}), (p_{-98,5} \to p_{-98,10}) \dots$$

dove usiamo la variabile $p_{i,j}$ con $i, j \in \mathbf{Z}$ per indicare che "$i < j$". Qui osserviamo un primo limite concreto della logica proposizionale. Per esprimere una quantificazione su un dominio infinito abbiamo bisogno di infinite proposizioni.

La situazione peggiora se consideriamo questo ragionamento:

> Ogni numero intero inferiore a 5 è inferiore a 10.
> 20 non è inferiore a 10
> Dunque esiste un numero non inferiore a 5

In questo caso la conclusione andrebbe espressa come una disgiunzione infinita del tipo $\bigvee_{i \in \mathbf{Z}} \neg p_{i,5}$. Ma il nostro formalismo non prevede disgiunzioni di questo tipo.

Esempio 1.2 (Revisited)
Immaginiamo che le proposizioni dell'Esempio 1 riguardino i dati di una "base di dati" contenenti informazioni sulle relazioni di amicizia di alcuni individui, tra i quali Pietro, Claudio e Marco. Il problema di una formalizzazione proposizionale è che il linguaggio dovrebbe contenere variabili $p_{a,b}$ per ogni coppia di individui a, b, con il significato intuitivo di a è amico di b. Se la base di dati venisse estesa con dati su nuovi individui, dovrei modificare di conseguenza il linguaggio. La natura *dinamica* di una base di dati rende opportuno passare a un formalismo diverso, in cui possiamo tenere fisso il linguaggio e valutare di volta in volta la verità delle proposizioni in questo linguaggio sui dati contenuti nella base di dati.

Vediamo come la logica predicativa risponde esattamente ai problemi che sono emersi dall'analisi degli esempi prededenti, ossia

(1) Permette di esprimere quantificazioni su dominii infiniti con singole proposizioni finite.
(2) Permette di distinguere le relazioni tra i concetti (proprietà) dalle relazioni tra gli individui.

Immaginiamo di avere una Base di Dati costituita da un insieme di individui, I, e da alcune tabelle che rappresentano relazioni tra questi individui. Per esempio ipotizziamo di avere una relazione $A \subseteq I \times I$ tale che $(i, j) \in A$ se e solo se i è amico di j; una relazione $F \subseteq I \times I$ tale che $(i, j) \in F$ se e solo se i è figlio di j; una relazione $S \subseteq I \times I$ tale che $(i, j) \in S$ se e solo se i è fratello/sorella di j. Supponiamo infine di saper distinguere individui maschi da individui femmine, ossia di avere accesso al sottinsieme $U \subseteq I$ contenente gli individui maschi e al sottinsieme $D \subseteq I$ contenente gli individui femmine.

Una base di dati di questo tipo è un oggetto *dinamico*, ossia è possibile aggiungere nuovi individui, e nuove coppie nelle relazioni di interesse. Se usassimo un linguaggio proposizionale per descrivere/interrogare la base di dati, dovremmo estendere il linguaggio ogni volta che si aggiungono nuovi dati. Questo non è conveniente e risulta più utile avere un linguaggio fisso.

Un linguaggio adeguato per descrivere/interrogare una base di dati come quella descritta sopra avrà le seguenti componenti. Un simbolo di relazione (predicato) per ogni relazione di interesse, dunque un simbolo per A, F, S; e un simbolo per i sottinsiemi U e D.

Usiamo le stesse lettere per indicare le relazioni e i simboli, ma attenzione, si tratta di due cose del tutto differenti: la relazione A è un insieme di coppie ordinate di elementi del dominio I; mentre il simbolo A è un simbolo formale in un linguaggio formale! La relazione A è l'interpretazione del simbolo A nella base di dati in questione.

Inoltre introduciamo dei nomi propri per alcuni (o per tutti) gli individui nel dominio: per es., un simbolo m per Marco, g per Gianni, ℓ per Laura.

Possiamo così scrivere proposizioni per esprimere concetti elementari quali: Marco è amico di Laura, Gianni è fratello di Marco, Laura non è figlia di Gianni: $A(m, \ell)$, $S(g, m)$, $\neg F(\ell, g)$.

Usando i connettivi booleani possiamo esprimere proposizioni più complesse, per es: se Marco è figlio di Laura, allora Gianni non è amico di Marco: $F(m, \ell) \to \neg A(g, m)$; oppure Marco e Laura sono figli di Gianni: $F(m, g) \wedge F(\ell, g)$, o ancora: Marco e Laura sono figli di Gianni, dunque sono fratelli: $((F(m, g) \wedge F(\ell, g)) \to S(m, \ell)$.

Immaginiamo ora di voler interrogare la base di dati per individuare chi sono gli amici di Marco, ossia l'insieme $\{i \in I \ : \ i \text{ è amico di Marco}\}$. Voglio scrivere una proposizione che risulti vera di tutti e soli gli amici di Marco. È naturale usare a questo scopo una variabile x, scrivendo:

$$\{x \in I \ \ A(x, m)\}.$$

L'idea è che l'insieme degli amici di Marco coincide con l'insieme dei valori $i \in I$ che rendono vera questa proposizione quando "sostituiti" alla variabile x. Per questo includo un insieme di variabili ($x, y, z, w,$ o $x_1, x_2, \dots$) al mio linguaggio, e considero un'espressione come $A(x, m)$ come una proposizione del mio linguaggio.

Se voglio individuare l'insieme degli amici degli amici di Marco, ossia:

$$\{i \in I \ : \ i \text{ è amico di un amico di Marco}\}$$

è naturale usare un quantificatore: gli amici degli amici di Marco sono gli individui $i \in I$ tali che esiste un individuo $j \in I$ tale che i è amico di j e j è amico di Marco. Posso scrivere:

$$\{x \in I \ : \ \exists y(A(y, m) \wedge A(x, y))\}$$

L'idea è che l'insieme

$$\{i \in I \ : \ i \text{ è amico di un amico di Marco}\}$$

coincide con l'insieme degli individui $i \in I$ che "rendono vera" la formula $\exists y(A(y, m) \wedge A(x, y))$ quando la variabile x viene interpretata come i. Considero dunque l'espressione $\exists y(A(y, m) \wedge A(x, y))$ una proposizione del linguaggio.

Potrei essere anche interessato a domande a risposta SI/NO, per esempio sapere se Marco ha figli. In questo caso posso scrivere la proposizione $\exists x(F(x, m))$; l'idea è di considerare vera questa proposizione formale se e solo se esiste un $i \in I$ tale che $F(x, m)$ è vera quanto x viene interpretato come i.

Per chiedere se Marco ha nipoti posso scrivere $\exists x \exists y(F(x, y) \wedge F(y, m))$.

Risulta spesso utile saper distinguere tra due individui differenti. A questo scopo introduciamo il simbolo $=$ nel linguaggio, da usarsi in espressioni del tipo $m = \ell$, $x = g$, $x = y$. Per esempio la seguente proposizione esprime il fatto che Marco ha almeno 2 figli: $\exists x \exists y(\neg(x = y) \wedge F(x, m) \wedge F(y, m)$.

Il linguaggio del nuovo tipo è detto linguaggio predicativo. Consta di simboli di **costanti**, **variabili**, **quantificatori** e **predicati**. Le costanti (che denoteremo per il momento con lettere latine minuscole $a, b, c, \dots$) vanno lette come nomi propri per individui del dominio di discorso. Le variabili (che denoteremo per il momento con $x, y, z, w, \dots$ o $x_1, x_2, \dots$) e i quantificatori (universale $\forall$ ed esistenziale $\exists$) permettono permettono di esprimere quantificazioni su domini finiti o infiniti. I simboli di predicato, che denoteremo con P, Q, R permettono di rendere esplicita la distinzione tra concetti (proprietà) e individui. I simboli di predicato vengono usati in combinazione con i simboli che denotano individui (variabili e costanti) in espressioni formali di tipo $P(x)$ o $P(c)$, da leggersi come "l'oggetto x ha la proprietà P", o "x è un P" (e analogamente per c). Accanto ai simboli di predicato usiamo i simboli di relazione, R, S, etc., in espressioni formali di tipo $R(x, y)$ o $S(c, y)$ (dove c è una costante).

Le proposizioni nel nuovo linguaggio sono di due tipi: le **proposizioni atomiche** sono di forma $R(t, s)$ o $(t = s)$, dove R è una relazione e t e s sono variabili o costanti.

Le proposizioni composte sono quelle ottenute dalle proposizioni atomiche applicando i connettivi booleani e i quantificatori. Per i connettivi booleani la definizione induttiva è identica a quella della logica proposizionale. Per i quantificatori poniamo: se A e B sono proposizioni e x è una variabile, allora le stringhe $((\forall x)A)$ e $((\exists x)A)$ sono proposizioni.

In conclusione, è opportuno includere nel linguaggio della logica predicativa i seguenti tipi di simboli.

(1) Costanti a, b, c, d, etc. (in quantità al più numerabile)
(2) Variabili x, y, z, w, etc., (in quantità numerabile)
(3) Quantificatori $\forall$, $\exists$,
(4) Simboli per predicati a uno o più posti $P(x)$, $R(x, y)$, $Q(x, y, z)$ etc., (in quantità al più numerabile).

Una scelta di simboli di predicato e di un insieme di costanti determina un *linguaggio predicativo relazionale.*

Torniamo ora ai nostri esempi problematici e formalizziamoli in un linguaggio del nuovo tipo (linguaggio predicativo).

Formalizzazione predicativa dell'Esempio 1.1, (versione 1). Prima di tutto dobbiamo scegliere un linguaggio opportuno. In particolare scegliere quanti predicati e quante costanti utilizzare. Possiamo analizzare la prima premessa come segue: Se un individuo ha la proprietà "Essere amico di Marco" allora quell'individuo ha la proprietà "Essere amico di Pietro". La seconda premessa di può analizzare come: L'individuo "Claudio" non ha la proprietà "Essere amico di Pietro". La terza premessa si può analizzare come: L'individuo "Claudio" non ha la proprietà "Essere amico di Marco". Questa analisi ci suggerisce che abbiamo bisogno di

(1) Un simbolo di costante c per l'individuo Claudio,
(2) Un simbolo di predicato AM per la proprietà Essere amico di Marco,
(3) Un simbolo di predicato AP per la proprietà Essere amico di Pietro.

Possiamo allora formalizzare l'argomento come segue.

$$\forall x(AM(x) \to AP(x))$$
$$\neg AP(c)$$
$$\neg AM(c)$$

Le regole della logica predicativa che svilupperemo saranno tali da rendere l'argomento logicamente valido (nel nuovo senso, da definire).

Formalizzazione predicativa dell'Esempio 1.1, (versione 2). Consideriamo ora la variante numerica dell'Esempio 1. Scegliamo un linguaggio opportuno. Possiamo analizzare la prima premessa come segue: Se un individuo ha la proprietà "Essere inferiore a 5" allora quell'individuo ha la proprietà "Essere inferiore a 10". La seconda premessa di può analizzare come: L'individuo "20" non ha la proprietà "Essere inferiore a 10". La terza premessa si può analizzare come: L'individuo "20" non ha la proprietà "Essere inferiore a 5". Questa analisi ci suggerisce che abbiamo bisogno di

(1) Un simbolo di costante v per l'individuo 20,
(2) Un simbolo di predicato $Inf5$ per la proprietà Essere inferiore a 5,
(3) Un simbolo di predicato $Inf10$ per la proprietà Essere inferiore a 10.

Possiamo allora formalizzare l'argomento come segue.

$$\forall x(Inf5(x) \to Inf10(x))$$
$$\neg Inf10(v)$$
$$\neg Inf5(v)$$

Abbiamo così ovviato al problema di formalizzare la quantificazione infinita con una proposizione finita. Si osserva facilmente che la versione formale appena proposta è identica alla versione formale dell'Esempio 1 proposta sopra a meno dei nomi dei predicati e delle costanti. I due argomenti sono infatti identici, ossia hanno la stessa forma logica.

Formalizzazione predicativa dell'Esempio 1.1, (versione 3). La formalizzazione dell'Esempio 1 e dell'Esempio 2 proposta qui sopra è corretta ma si può migliorare. Nell'Esempio 1 è chiaro che i predicati "Essere amico di Marco" e "Essere amico di Pietro" hanno qualcosa in comune, e lo stesso vale per i predicati "Essere inferiore a 5" e "Essere inferiore a 10" nell'Esempio 2. È naturale che questa comunanza venga rispecchiata nel nostro linguaggio formale. A questo scopo estendiamo il concetto di predicato a quello di "predicato a più variabili", o "relazione" e introduciamo nel linguaggio simboli per predicati di questo nuovo tipo. Un'espressione di tipo $P(x, y)$ sarà letta come gli oggetti x, y stanno nella relazione P. Il nostro linguaggio può ora contenere simboli di predicato con qualunque numero finito di variabili. Un'espressione di tipo $R(x_1, \ldots, x_n)$ sarà letta come "Gli individui indicati da $x_1, \ldots, x_n$ stanno nella relazione R". Possiamo allora proporre una nuova formalizzazione degli Esempi 1 e 2 usando il seguente linguaggio.

(1) Un simbolo di predicato $P(x, y)$ a due posti,
(2) Tre simboli di costante a, b, c.

Per l'Esempio 1 $P(x, y)$ viene letto come "x è amico di y", a come Marco, b come Pietro e c come Claudio. Per l'Esempio 2 $P(x, y)$ viene letto come "x è inferiore a y", a come 5, b come 10 e c come 20. La versione formalizzata dei due Esempi è la seguente.

$$\forall x(P(x, a) \to P(x, b))$$
$$\neg P(c, b)$$
$$\neg P(c, a)$$

Osservazione 1.2. Abbiamo discusso sopra alcuni esempi numerici e abbiamo scelto di introdurre costanti (nomi propri) per denotare singoli numeri. Questo è corretto ma si può fare di meglio. Da una parte è poco economico introdurre infiniti nomi propri distinti. Dall'altra, usare una costante c per il numero 20 e una costante a per il numero 21 nasconde la relazione strutturale tra 20 e 21, ossia il fatto che il secondo si ottiene dal primo aggiunge 1. In fin dei conti è ovvio che tutti i numeri naturali maggiori di 0 si possono scrivere come somme di 1! Potrebbe essere utile rispecchiare questa struttura nel nostro nuovo linguaggio. A questo scopo introduciamo un nuovo tipo di simboli, i simboli di funzione. Come i predicati, questi simboli possono avere uno, due, tre, o n posti (argomenti). A differenza dei predicati, che si usano in espressioni del tipo $P(x)$ per asserire una proposizione sensata del tipo "x è un P", i simboli di funzione si usano per formare *nuovi nomi propri*. Per esempio, il numero 4 può essere indicato come $1 + 1 + 1 + 1+$, o come $2 + 2$ o come $2 \cdot 2$, o come 2^2. Questi quattro nomi propri sono formati da costanti (1, 2) e funzioni ($+$, $\cdot$, x^y). A livello formale introdurre nel linguaggio un simbolo di funzione per rappresentare, e.g., l'addizione, significa specificare un simbolo, per esempio "p" da usare in espressioni del tipo "$p(c, y)$", dove i due argomenti del simbolo di funzione p possono essere costanti (come c) o variabili come y. L'espressione $p(c, y)$ va letta come "Il risultato di applicare la funzione denotata dal simbolo p agli individui denotati dalla costante c e dalla variabile y". Un'espressione ottenuta applicando un simbolo di funzione a variabili o costanti è detta un **termine**. Un termine non contiene variabili è detto un **termine chiuso**. I termini chiusi sono da intendere come nomi propri composti per individui del dominio del discorso. L'interpretazione di tali termini è fissata una volta che è fissata l'intepretazione delle costanti e dell'operazione associata al simbolo di funzione. Ammettere variabili nei termini è una convenienza tecnica.

Formalizzazione predicativa dell'Esempio 1.1, (versione 4). Tornando ai nostri esempi numerici, è naturale usare un linguaggio con due soli costanti 0 e 1 e un simbolo di operazione p a due posti che formalizza la somma. Possiamo allora scrivere 2 come $p(1, 1)$, 3 come $p(1, p(1, 1))$, 4 come $p(1, p(1, p(1, 1)))$ o anche come $p(p(1, 1), p(1, 1))$, e così via. Spesso scriviamo $x + y$ invece di $p(x, y)$, per migliorare la leggibilità (a costo di favorire qualche confusione!). Inoltre, per non scrivere per esteso lunghe somme formali, usiamo $\overline{n}$ per indicare il termine ottenuto applicando alla costante 1 il simbolo di funzione p per n volte!

Con questo nuovo linguaggio l'Esempio 1 si formalizza come segue.

$$\forall x(P(x, \overline{5}) \to P(x, \overline{10}))$$
$$\neg P(\overline{20}, \overline{10})$$
$$\neg P(\overline{20}, \overline{5})$$

Esempio 1.3. Consideriamo il seguente argomento.

Ogni multiplo di 100 è multiplo di 10.
25 non è multiplo di 10.
Dunque 25 non è multiplo di 100.

Potremmo formalizzare questo esempio esattamente come abbiamo fatto per l'esempio precedente, formalizzando il concetto "Essere multiplo di" con un simbolo di relazione a due posti $P(x, y)$. Si può fare di meglio. Ovviamente il concetto "Essere multiplo di" si può analizzare ulteriormente. Abbiamo infatti che, per $m, n \in \mathbf{N}$ vale che m è multiplo di n se e soltanto se esiste $a \in \mathbf{N}$ tale che $m = n \cdot a$. Possiamo esprimere nel nostro nuovo linguaggio questa condizione, come segue, introducendo un nuovo simbolo di funzione, *mult* a due posti, e un nuovo simbolo $=$ per indicare l'identità.

$$\exists z(x = mult(y, z)).$$

L'argomento si formalizza allora come segue.

$$\begin{gathered}\forall x(\exists z(x = mult(\overline{100}, z)) \to \exists z(x = mult(\overline{10}, z))) \\ \neg\exists z(25 = mult(\overline{10}, z)) \\ \neg\exists z(25 = mult(\overline{100}, z))\end{gathered}$$

In conclusione, è opportuno includere nel linguaggio della logica predicativa i seguenti tipi di simboli.

(1) Costanti a, b, c, d, etc.,
(2) Variabili x, y, z, w, etc.,
(3) Quantificatori $\forall$, $\exists$,
(4) Simboli per predicati a uno o più posti $P(x)$, $R(x, y)$, $Q(x, y, z)$ etc.,
(5) Simboli per funzioni a uno o più posti $p(x)$, $f(x, y)$, $g(x, y, z)$, etc.,
(6) Un simbolo per l'identità, $=$.

Osservazione 1.4. Dato che l'identità è una relazione a due posti e le funzioni a n-argomenti sono completamente descritte dal loro grafico che è una relazione a $n + 1$ posti, si potrebbe fare a meno di avere simboli speciali per questo tipo di oggetti. È però naturale avere simboli di funzione nel linguaggio ed è opportuno avere il simbolo di identità da interpretare come l'identità nel dominio di discorso (per motivi che potremmo discutere solo più avanti).

Per formalizzare proposizioni del linguaggio naturale nel linguaggio predicativo formale si possono seguire alcune regole guida per le forme più comuni di proposizioni.

Per esprimere una proposizione di tipo "Ogni A è B" si usa il formato $\forall x(A(x) \to B(x))$. Per esempio: Ogni amico di Marco è amico di Laura si traduce in $\forall x(A(x, m) \to A(x, \ell))$.

Per esprimere una proposizione di tipo "Qualche A è B" si usa il formato $\exists x(A(x) \wedge B(x))$; per esempio: qualche figlio di Laura è figlio di Gianni, si esprime con la formula $\exists x(F(x, \ell) \wedge F(x, g))$.

Per esprimere una proposizione di tipo "Nessun A è B" si può usare la formula $\forall x(A(x) \to \neg B(x))$ (ogni A non è B) oppure $\neg\exists x(A(x) \wedge B(x))$ (non esiste un A che sia anche un B. Per esempio per esprimere: Nessun fratello di Gianni è fratello di Laura possiamo usare $\forall x(S(x, g) \to \neg S(x, \ell))$ oppure $\neg\exists x(S(x, g) \wedge S(x, \ell))$.

LOGICA MATEMATICA

A.A. 23/24, LOGICA PREDICATIVA, DISPENSA N. 2

SOMMARIO. Introduciamo il linguaggio, la sintassi e la semantica della Logica dei Predicati.

1. STRUTTURE

I costrutti sintattici (relazioni, funzioni, costanti) emersi nella discussione precedente possono essere giustificati anche come i costrutti minimi necessari a descrivere le proprietà di una generica struttura matematica. La nozione di struttura è ubiqua nella Matematica moderna. Cominciamo con qualche esempio.

Esempio 1.1. Un gruppo è definito da un insieme A (gli elementi del gruppo), un elemento $e \in A$ (l'elemento neutro) e un'operazione binaria $\circ : A \times A \to A$, che soddisfa le seguenti proprietà:

(1) Per ogni $a \in A$, $a \circ e = e \circ a = a$
(2) Per ogni a inA esiste un $b \in B$ 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 abeliano se inoltre soddisfa: per ogni $a, b \in A$ si ha $a \circ b = b \circ a$.

Esempio 1.2. Un grafo è definito da un insieme V di vertici e una relazione $E \subseteq V \times V$ di adiacenza.

Esempio 1.3. Un ordine parziale è definito come un insieme X di elementi e una relazione binaria $\leq \subseteq X \times X$ con le seguenti proprietà:

(1) Per ogni $x \in X$: $x \leq x$
(2) Per ogni $x, y \in X$: se $x \leq y$ e $y \leq x$ allora $x = y$.
(3) Per ogni $x, y, z \in X$, se $x \leq y$ e $y \leq z$ allora $x \leq z$.

Esempio 1.4. Un anello è definito da un insieme A, un elemento $e \in A$, due operazioni binarie, $+$ e $\times$, che soddisfano le seguenti proprietà:

(1) A è un gruppo abeliano relativamente all'operazione $+$ e l'elemento neutro è e.
(2) Per ogni $a, b, c \in A$: $a \times (b \times c) = (a \times b) \times c$.
(3) Per ogni $a, b, c \in A$: $a \times (b + c) = (a \times b) + (a \times c)$.
(4) Per ogni $a, b, c \in A$: $(a + b) \times c = (a \times c) + (b \times c)$.

E così via per altre strutture note (e.g., campi, campi ordinati, etc.)
Cosa hanno di comune gli esempi qui sopra? Grafi, gruppi, ordini parziali etc. sono presentati indicando:

(1) Un insieme di **elementi** (tipicamente non vuoto) A.
(2) Una o più **relazioni** su A (anche unarie – ossia sottinsiemi di A).
(3) Una o più **funzioni** su A.
(4) Una o più **costanti** in A (elementi di A con particolari proprietà).

Chiamiamo *struttura* un oggetto di questo genere. Una volta fissata una struttura di interesse, siamo abituati ad esplorarne le proprietà formulando enunciati veri nella struttura, mediante connettivi logici booleani o quantificazioni sugli elementi della struttura.

La nozione di Linguaggio Predicativo che definiremo è ideato esattamente avendo in mente questo concetto di struttura.

Note preparate da Lorenzo Carlucci, lorenzo.carlucci@uniroma1.it.

2. Sintassi della Logica Predicativa

Definizione 2.1 (Linguaggio predicativo)**.** Linguaggio predicativo $\mathcal{L}$ è una collezione (finita o infinita) di simboli di tre tipi.

- *Simboli di relazioni*, ciascuno con la sua molteplicità (o arietà).
- *Simboli di funzioni*, ciascuno con la sua molteplicità (o arietà).
- *Simboli di costanti*.

Inoltre assumiamo sempre un insieme numerabile di variabili $v_1, v_2, \ldots$.

Inoltre assumiamo sempre un insieme numerabile di variabili $v_1, v_2, \ldots$. Queste sono le variabili ufficiali del linguaggio. Useremo x, y, z, w, v (con pedici) come variabili su variabili (dette anche metavariabili, perché sono usate come variabili per variabili individuali). Costanti e simboli di funzione si possono combinare per costruire nomi più complessi per elementi del dominio.

Esempio 2.2. Un linguaggio adeguato per la teoria dei gruppi è il seguente $\mathcal{L}_G = \{\cdot, e\}$; dove $\cdot$ è un simbolo di funzione binaria per l'operazione di gruppo e e una costante per l'elemento neutro.

Un linguaggio per l'aritmetica è il seguente $\mathcal{L}_A = \{+, \times, <, 0, 1\}$, dove $+, \times$ sono simboli di funzioni a due argomenti, $<$ un simbolo di relazione binario e $0, 1$ sono costanti.

Un linguaggio per la teoria degli insiemi è il seguente $\mathcal{L}_I = \{\in\}$ dove $\in$ è un simbolo di relazione binario.

Costanti e funzioni si possono combinare per costruire nomi più complessi per elementi del dominio. Per esempio, $1 + (1 + (1 + 1))$ e $(1 + 1) \times (1 + 1)$ sono due nomi per il numero 2. Analogamente siamo abituati a manipolare espressioni come x^2, $x \times x$, $x + 1$, etc.

Definizione 2.3 (Termini)**.** I *termini* sono ottenuti partendo dalle variabili e dalle costanti e chiudendo sotto applicazione di simboli di funzione. Un termine che non contiene variabili è un *termine chiuso*.

Esempio 2.4. $((v_9 + 1) + 1) \times 0$ è un termine nel linguaggio $\mathcal{L}_A$. e, $e \cdot e$, $e \cdot (e \cdot e)$, $(e \cdot e) \cdot e$ sono termini chiusi nel linguaggio $\mathcal{L}_G$. I termini del linguaggio $\mathcal{L}_I$ sono solo le variabili libere. N.B. Spesso usiamo la notazione infissa (i.e., $(x + y)$) invece di quella prefissa (i.e., $+(x, y)$).

Per formulare proposizioni nel linguaggio $\mathcal{L}$ usiamo i simboli logici seguenti

- Connettivi $\wedge$, $\vee$, $\rightarrow$, $\neg$.
- Quantificatori $\exists$, $\forall$.
- Il simbolo di eguaglianza o identità $=$.

In logica proposizionale le variabili proposizionali erano la minima unità dotata di un valore di verità (0,1). In logica predicativa il loro posto viene preso da espressioni più complesse, dette formule atomiche.

Definizione 2.5 (Formule Atomiche)**.** Una formula atomica è una stringa del tipo $R(t_1, \ldots, t_k)$ dove R è un simbolo di relazione a k posti e $t_1, \ldots, t_k$ sono termini; o una stringa del tipo $(t = s)$ dove t, s sono termini.

Esempio 2.6. In un linguaggio composto da due costanti c_0 e c_1 e da due simboli di predicato P (a un posto) e R (a due posti), le sole formule atomiche sono del tipo $P(c_0)$, $P(c_1)$, $P(x)$, $R(x, c_1)$, $R(c_0, c_1)$, etc. Le sole formule atomiche *senza variabli* sono $P(c_0)$, $P(c_1)$, $R(c_0, c_1)$, $R(c_1, c_0)$, $R(c_0, c_0)$, $R(c_1, c_1)$.

Definizione 2.7 (Formule)**.** Le formule sono ottenute partendo dalle formule atomiche e chiudendo sotto connettivi proposizionali e quantificatori universali ed esistenziali. Le formule (non atomiche) sono dunque del tipo

$$(F \wedge G), (F \vee G), (\neg F), (F \rightarrow G), ((\forall v)F), ((\exists v)F),$$

dove F e G sono formule (atomiche o non atomiche) e v è una variabile.

Nelle formule $((\forall v)F)$ e $((\exists v)F)$, F è detto il *dominio* (o *scope*, in inglese) del quantificatore e v la variabile quantificata. Se v non occorre in F possiamo identificare le due formule quantificate con F. Una distinzione fondamentale è quella tra variabili quantificate (dette vincolate o legate) e variabili non quantificate (dette libere).

Definizione 2.8 (Variabili libere e legate)**.** Una occorrenza di una variabile x in una formula F è vincolata se e solo se (i) l'occorrenza di x è la variabile quantificata di un quantificatore, oppure (ii) l'occorrenza di x è nel dominio di un quantificatore con variabile quantificata x. Tutte le altre occrrenze di x in F sono dette libere.

Ad ogni formula F possiamo associare in modo ovvio l'insieme delle sue variabili libere (le variabili che hanno almeno un'occorrenza libera in F) e l'insieme delle sue variabili legate (le variabili che hanno almeno un'occorrenza vincolata in F). Ovviamente i due insiemi non sono necessariamente disgiunti.

Esempio 2.9. Le variabili quantificate funzionano in modo simili alle variabili locali in un linguaggio di programmazione. Nella formula

$$(\forall x R(x,y)) \wedge (\exists y S(x,y,z))$$

la prima occorrenza di x (quella in $R(x,y)$ è legata (dal quantificatore $\forall x$, la prima occorrenza di y è libera, la seconda occorrenza di y è legata (dal quantificatore $\exists y$) e la seconda occorrenza di x è libera. La variabile z ha una sola occorrenza ed è libera.

Un *enunciato* è una formula senza variabili libere.

In alcuni casi è utile sostituire una occorrenza di una variabile x in una formula F con un termine t. La prossima definizione indica in quali casi questa sostituzione è legittima. Il succo è che se sostituiamo t per x in $F(x)$ nessuna occorrenza di una variabile in t diventa vincolata in $F(t)$.

Definizione 2.10. 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.

Esempio 2.11. Se la formula F è $\exists y(x = y + y)$, allora nessun termine contenente y è libero per x in F.

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

3. Semantica della Logica Predicativa

Il nostro scopo è di definire la nozione di *verità logica* per la logica dei predicati. Questa nozione è analoga a quella di tautologia per la logica proposizionale, ossia indica una formula sempre vera, qualunque sia l'assegnamento di un significato ai simboli che la compongono. A tale scopo dobbiamo definire la nozione di verità di una formula della logica predicativa. Nel caso della logica proposizionale la verità di una formula è fissata una volta che è fissato un assegnamento di valori di verità alle variabili proposizionali. Nel caso della logica predicativa la verità di una proposizione dipende dalla scelta dell'ambiente in cui decidiamo di interpretare i simboli del linguaggio. Un tale ambiente è detto *struttura*.

Si tratta di astrarre dalla normale pratica matematica. In Algebra è abituale definire un gruppo come un insieme G su cui è definita una operazione $\cdot$ associativa e tale che esiste un elemento e di G che è neutro rispetto a $\cdot$ e commuta con tutti gli elementi di G. In Teoria dei Grafi è abituale definire un grafo (semplice) G come una coppia (V, E), dove V è un insieme (detto insieme dei vertici) e E è una relazione binaria non riflessiva e simmetrica (i.e., gli archi non hanno orientazione). Gruppi e grafi sono esempi di *strutture*, ossia insiemi su cui sono definite operazioni, relazioni e costanti.

In Algebra ci si può interessare alle proprietà di *singoli gruppi* di particolare interesse, o alle proprietà di *classi di gruppi* di particolare interesse (per esempio i gruppi abeliani, i gruppi ciclici, i gruppi di permutazioni, etc.), o alle proprietà di *tuttti i gruppi*. In Teoria dei Grafi ci si può interessare alle verità che valgono in singoli grafi di particolare interesse, o alle verità che valgono di classi di grafi di particolare interesse (bipartiti, completi, Euleriani, etc.), o infine alle verità che valgono per tutti i grafi. In Logica Matematica facciamo un passo di generalizzazione ulteriore e ci interessiamo alla verità *in tutte le strutture*. Per questo motivo diamo le seguenti definizioni in forma molto generale.

Definizione 3.1. Fissiamo un linguaggio $\mathcal{L} = \{R_i, f_j, c_k \ : \ i \in I, k \in K\}$ dove I, J, K sono insiemi (di indici). I simboli f_j sono simboli di funzione, i simboli R_i sono simboli di relazione; i simboli c_k sono simboli di costante. Una struttura (o interpretazione) $\mathfrak{A}$ 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 $R_i^{\mathfrak{A}}$, dove $R_i^{\mathfrak{A}} \subseteq A^d$.
- Per ogni simbolo f_j di arietà d una funzione a d argomenti sul dominio A, che denotiamo con $f_j^{\mathfrak{A}}$, dove $f_j^{\mathfrak{A}} : A^d \to A$.
- Per ogni $k \in K$, un elemento di A, che denotiamo con $c_k^{\mathfrak{A}}$.

Esempio 3.2. Se il linguaggio contiene soltanto le costanti c_0, c_1 e i simboli P (a un posto) e R (a due posti), una struttura per il linguaggio è determinata da:

(1) Un insieme non vuoto, A.
(2) Un elemento di A per interpretare c_0, che denotiamo con $c_0^{\mathfrak{A}}$, e un elemento di A per interpretare c_1, che denotiamo con $c_1^{\mathfrak{A}}$.
(3) Un sottinsieme di A per interpretare P; che denotiamo con $P^{\mathfrak{A}}$.
(4) Una relazione binaria su A, ossia un sottinsieme di $A \times A$; che denotiamo con $R^{\mathfrak{A}}$.

Per esempio, una struttura possibile $\mathfrak{A}$ per questo linguaggio si ottiene scegliendo come dominio $\mathbf{N}$, come $c_0^{\mathfrak{A}}$, $c_1^{\mathfrak{A}}$, $R^{\mathfrak{A}} = <$, e $P^{\mathfrak{A}} =$ i numeri pari. Un'altra si ottiene scegliendo come dominio gli interi, come interpretazione di c_0 il numero 0, come interpretazone di c_1 il numero -1, come interpretazione di R la relazione $\geq$ sugli interi.

Definiamo la relazione di *validità di una formula F in una struttura* $\mathfrak{A}$, che denotiamo con $\mathfrak{A} \models F$.

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

$$\alpha : \{v_n \,:\, n \in \mathbf{N}\} \longrightarrow A$$

Un assegnamento si estende in modo univoco ai termini ponendo $\alpha(c)$ uguale a $c^{\mathfrak{A}}$ (ossia: l'interpretazione di una costante in una struttura è, per l'appunto, costante).

Osservazione 3.3. Un assegnamento si estende univocamente ai termini ponendo $\alpha(c)$ uguale a $c^{\mathfrak{A}}$ e $\alpha(f(t_1, \ldots, t_k))$ uguale a $f^{\mathfrak{A}}(\alpha(t_1), \ldots, \alpha(t_k))$.

Indichiamo con $\alpha\binom{x}{a}$ l'assegnamento che differisce da α solo perché associa l'elemento a alla variabile x.

Definiamo la relazione $\mathfrak{A} \models F[\alpha]$, che intuitivamente significa: la formula F è soddisfatta nella struttura $\mathfrak{A}$ relativamente all'assegnamento α.

La seguente nozione di verità di una formula in una struttura relativamente a un assegnmaneto è dovuta ad Alfred Tarski.

Definizione 3.4 (Soddisfazione)**.** Definiamo la relazione $\mathfrak{A} \models F[\alpha]$ come segue, per induzione sulla complessità di F.

$\mathfrak{A} \models R(t_1, \ldots, t_k)[\alpha]$ se e solo se $(\alpha(t_1), \ldots, \alpha(t_k)) \in R^{\mathfrak{A}}$, dove $t_1, \ldots t_k$ sono *termini* e R è un simbolo di relazione di arietà k.

$\mathfrak{A} \models (t = s)[\alpha]$ se e solo se $\alpha(t) = \alpha(s)$, dove t e t sono *termini*.

$\mathfrak{A} \models \neg G[\alpha]$ se e solo se non vale $\mathfrak{A} \models G[\alpha]$.

$\mathfrak{A} \models (G \wedge H)[\alpha]$ se e solo se $\mathfrak{A} \models G[\alpha]$ e $\mathfrak{A} \models H[\alpha]$.

$\mathfrak{A} \models (G \vee H)[\alpha]$ se e solo se $\mathfrak{A} \models G[\alpha]$ o $\mathfrak{A} \models H[\alpha]$.

$\mathfrak{A} \models (G \to H)[\alpha]$ se e solo se: se $\mathfrak{A} \models G[\alpha]$ allora $\mathfrak{A} \models H[\alpha]$.

$\mathfrak{A} \models (\exists v G)[\alpha]$ se e solo se esiste $a \in A$ tale che $\mathfrak{A} \models G[\alpha\binom{v}{a}]$.

$\mathfrak{A} \models (\forall v G)[\alpha]$ se e solo se per ogni $a \in A$ vale $\mathfrak{A} \models G[\alpha\binom{v}{a}]$.

Esempio 3.5. Sia $\mathcal{L} = \{c_0, c_1, P, R\}$ con P a un posto e R a due posti. Sia $F(x)$ la formula $\exists y(R(x, y)$. Consideriamo la struttura $\mathcal{N}$ con dominio $\mathbf{N}$ che interpreta la costante c_0 nel numero 0 la costante c_1 nel numero 1, il simbolo P con l'insieme dei numeri pari, il simbolo R con la relazione $<$ (la normale relazione d'ordine stretto sui numeri naturali). Svolgendo la definizione abbiamo: $\mathcal{N} \models F(x)[\alpha]$ se e solo se esiste $n \in \mathbf{N}$ tale che $\mathcal{N} \models R(x, y)[\alpha\binom{y}{n}]$ se e solo se $\alpha(x)$ è minore stretto di n. Dunque $\mathcal{N} \models F(x)[\alpha]$ se e solo se $\alpha(x)$ è maggiore di 0.

Osservazione 3.6. Il fatto che valga $\mathfrak{A} \models F[\alpha]$ o no dipende soltanto dai valori di α sulle variabili libere che appaiono in F. In altre parole, se le variabili libere di F sono contenute in $\{x_1, \ldots, x_n\}$ e α e β sono due assegnamenti che coincidono sui valori assegnati alle variabili $x_1, \ldots, x_n$, allora $\mathfrak{A} \models F[\alpha]$ se e solo

se $\mathfrak{A} \models F[\beta]$. Pertanto possiamo scrivere $\mathfrak{A} \models F[\alpha(x_1), \ldots, \alpha(x_n)]$ indicando esplicitamente gli elementi assegnati alle variabili che contano. Da questa osservazione segue anche che se F è un enunciato, allora $\mathfrak{A} \models F[\alpha]$ vale per tutti gli assegnamenti o per nessuno!

Esempio 3.7. Sia $\mathcal{L} = \{c, R\}$ dove R un simbolo di relazione binaria. Sia F l'enunciato $\forall x \forall y(R(x, y) \to \exists z(R(x, z) \wedge R(z, y)))$. Sia $\mathfrak{A}$ la struttura con dominio $\mathbb{Z}$, che interpreta c in 0 e R nella relazione d'ordine $<$. $\mathfrak{A} \models F[\alpha]$ se e solo se per ogni intero a se $R(\alpha(x), a)$ allora per esiste un intero b tale che $a < b$ e $b < a$. L'enunciato risulta dunque falso in $\mathbb{Z}$, ossia non vale $\mathbb{Z} \models F$. Se cambiamo struttura e consideriamo quella con dominio $\mathbf{Q}$ (i razionali), l'enunciato risulta vero (corrisponde al fatto che i razionali sono ordinati in modo denso).

Definizione 3.8 (Soddisfacibilità, Validità in una struttura)**.** Se $\mathfrak{A} \models F[\alpha]$ per qualche assegnamento α, diciamo che α *soddisfa* l'enunciato F in $\mathfrak{A}$, e in tal caso F è detta *soddisfacibile in* $\mathfrak{A}$. Diciamo che una formula F è *vera in una struttura* se è soddisfatta da tutti gli assegnamenti su quella struttura. In questo caso scriviamo $\mathfrak{A} \models F$.

Osservazione 3.9. Una formula F è vera in una struttura se e solo se l'enunciato

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

è vero nella struttura, dove $x_1, \ldots, x_n$ sono tutte e sole le variabili libere di F

Di fatto ci basta ragionare sulla validità di enunciati (ossia formule senza variabili libere). In ogni struttura data, un enunciato è soddisfatto da tutti gli assegnamenti o da nessuno.

Definizione 3.10 (Validità)**.** Un enunciato E è *valido* se è vero in tutte le strutture, ossia

$$\text{per ogni } \mathfrak{A} \text{ vale } \mathfrak{A} \models E$$

Diciamo anche che è una *verità logica* e scriviamo $\models E$.

Dualmente un enunciato E è *insoddisfacibile* se non esiste nessuna strtuttura in cui è vero, ossia

$$\text{per ogni } \mathfrak{A} \text{ non vale } \mathfrak{A} \models E.$$

Si osserva che potremmo estendere la nozione di verità logica appena definita anche a formule (con variabili libere).

Esempi di verità logiche sono ereditati dalla Logica Proposizionale: per esempio si consideri l'enunciato $\forall x R(x) \vee \neg \forall x R(x)$. Si verifica facilmente che questo enunciato è soddisfatto in ogni struttura (adeguata, ossia che interpreti il simbolo R). Dunque $\models \forall x R(x) \vee \neg \forall x R(x)$. L'enunciato è ottenuto dalla tautologia proposizionale $p \vee \neg p$ sostituendo alla variabile p l'enunciato $\forall x R(x)$. Sostituendo a p un qualunque altro enunciato otterrei analogamente una verità logica, per esempio $\forall x \exists y R(x, y) \vee \neg \forall x \forall y R(x, y)$. Enunciati ottenuti in questo modo da tautologie proposizionali vengono detti *istanze* di tautologie proposizionali. Si osserva facilmente che ogni istanza di una tautologia proposizionale è una verità logica (predicativa).

Vi sono poi verità logiche predicative che non si ottengono in questo modo. Esse riguardano il comportamento dei quantificatori. Per esempio abbiamo verificato usando la nozione di soddisfazione che l'enunciato $\exists y \forall x R(x, y) \to \forall x \exists y R(x, y)$ è vero in ogni struttura. Indagheremo più sistematicamente lo spazio delle verità logiche predicative.

LOGICA MATEMATICA

A.A. 23/24, LOGICA PREDICATIVA, DISPENSA N. 3

SOMMARIO. Teorie, e conseguenza logica. Problemi generali. L'ordine sui razionali. Isomorfismo tra modelli numerabili.

1. TEORIE

Una *teoria* in un linguaggio $\mathcal{L}$ è un insieme T di enunciati in $\mathcal{L}$.

Un *modello* di una teoria T è una struttura per $\mathcal{L}$ che soddisfa tutti gli elementi di T. In questo caso scriviamo $\mathfrak{A} \models T$.

Una teoria è *soddisfacibile* se ha un modello.

Diciamo che T *implica logicamente* E se E è vero in tutti i modelli di T. In questo caso scriviamo $T \models E$.

Siamo interessati all'insieme delle conseguenze logiche di una teoria T, che definiamo come segue:

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

Un'altra domanda naturale, data una teoria T, è quali siano i suoi modelli. Definiamo

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

Infine possiamo essere interessati alle proprietà di una specifica struttura matematica $\mathfrak{A}$ (e.g., i numeri naturali con somma e prodotto). In questo caso, se $\mathfrak{A}$ è una struttura, definiamo la teoria di $\mathfrak{A}$ come segue

$$Th(\mathfrak{A}) = \{E : \mathfrak{A} \models E\}.$$

Esempio 1.1. Un esempio di teoria è la teoria formata dagli usuali assiomi di gruppo scritti in un linguaggio predicativo $\mathcal{L}_G$ il linguaggio $\mathcal{L}_{\text{Gruppi}}\{\circ, e\}$, dove $\circ$ è un simbolo di funzione a due posti e e un simbolo di costante. Gli assiomi di gruppo si esprimono naturalmente con i seguenti enunciati:

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

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

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

L'insieme dei tre enunciati precedenti forma la teoria dei gruppi, che indichiamo con T_{Gruppi}.

I modelli della teoria T_{Gruppi} sono esattamente le strutture che chiamiamo gruppi (in questo caso la teoria è usata per definire la nozione). Dunque $Mod(T_{\text{Gruppi}})$ = l'insieme di tutti e soli i gruppi.

In Algebra, quando studiamo le proprietà dei gruppi, studiamo gli enunciati che sono veri in qualunque gruppo ossia in ogni struttura che soddisfa gli assiomi di gruppo. La veste formale di questa idea è l'insieme delle conseguenze logiche della teoria T_{Gruppi}, ossia $\{E : T_{\text{Gruppi}} \models E\}$, dove si sottintende che E varia sugli enunciati nel linguaggio $\mathcal{L}_{\text{Gruppi}}\{\circ, e\}$, della teoria.

D'altra parte possiamo essere interessati alle proprietà di un singolo gruppo. In questo caso, se $\mathcal{G}$ è il gruppo in questione, siamo interessati all'insieme $\{E : \mathcal{G} \models E\}$, che chiamiamo *la teoria di* $\mathcal{G}$. Per esempio, possiamo essere interessati a $Th()$.

Per gli insiemi sopra definiti $Conseq(T)$ e $Th(\mathfrak{A})$, associati naturalmente a una teoria T o a una struttura $\mathfrak{A}$, è anche interessante chidersi se esista un algoritmo che decide l'appartenenza di un enunciato a tali insiemi. Per esempio se ci chiediamo se esiste un algoritmo per decidere l'appartenenza di un enunciato a $Conseq(T_{\text{Gruppi}})$ ci stiamo chiedendo se le conseguenze degli assiomi di gruppo siano riconoscibili meccanicamente/algoritmicamente.

Appunti preparati da Lorenzo Carlucci, lorenzo.carlucci@di.uniroma1.it.

Si osserva facilmente che la teoria di una singola struttura $\mathfrak{A}$ è sempre completa nel senso che per ogni enunciato E, o vale $\mathfrak{A} \models E$ o vale $\mathfrak{A} \models \neg E$.

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, è facile vedere che le conseguenze degli assiomi di gruppo (ossia della teoria T_{Gruppi}) non è un insieme completo: dato che esistono gruppi abeliani e gruppi non-abeliani e dato che la proprietà di essere abeliano è agevolmente esprimibile nel linguaggio dei gruppi con una enunciato predicativo ($C = \forall x \forall y (x \cdot y = y \cdots x)$), 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 $Conseq(T_{\text{Gruppi}}$ (si legga: né l'essere abeliano né l'essere non abeliano sono conseguenze degli assiomi di gruppo). La teoria T_{Gruppi} è dunque incompleta.

2. La teoria dell'ordine dei razionali

Consideriamo i razionali con la relazione d'ordine $\leq$, ossia la struttura $\mathcal{Q} = (\mathbb{Q}, \leq)$. Si tratta di una struttura adeguata per un linguaggio $\mathcal{L}$ contenente un singolo simbolo di relazione binaria R.

Se volessimo descrivere le proprietà peculiari che distinguono $\mathcal{Q}$ da altre strutture ordinate, diremmo naturalmente che si tratta di un insieme con un ordine totale (riflessivo, antisimmetrico, transitivo e totale), senza estremi (per distinguerlo da un ordine finito), denso (per distinguerlo dall'ordine su $\mathbb{N}$ e $\mathbb{Z}$) e su insieme numerabile (per distinguerlo dall'ordine $\leq$ su $\mathbb{R}$).

Tutte le proprietà qui sopra, a eccezione della proprietà di essere numerabile, si traducono immediatamente in enunciati predicativi nel linguaggio $\mathcal{L} = \{R(x, y)\}$. Per leggibilità useremo il simbolo $\leq$ invece di R. Otteniamo così i seguenti enunciati, dove usiamo $x < y$ come abbreviazione di $(x \leq y \wedge \neg(x = y)))$, che esprime in modo naturale il concetto di ordine lineare denso senza estremi.

(1) (A1 - Riflessività) $\forall x (x \leq x)$,
(2) (A2 - Transitività) $\forall x \forall y \forall z (x \leq y \wedge y \leq z \to x \leq z)$,
(3) (A3 - Antisimmetria) $\forall x \forall y ((x \leq y \wedge y \leq x) \to y = x)$,
(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) $\forall x \forall y (x < y \to \exists z (x < z \wedge z < y))$.

Chiamiamo DLO (Dense Linear Order) questo insieme di enunciati, ossia questa teoria.

Ci chiediamo adesso se le proprietà sopra elencate, insieme alla proprietà di avere cardinalità numerabile, descrivono $(\mathbb{Q}, \leq)$ nel modo più stringente possibile. Come usuale in Matematica questo significa *a meno di isomorfismo*. Ci chiediamo cioè se ogni struttura $(A, \preceq)$ dove A è un insieme numerabile e $\preceq$ un ordine totale denso e senza estremi su A sia isomorfa a $(\mathbb{Q}, \leq)$.

In questo contesto è naturale definire l'isomorfismo come segue: $(A, \preceq)$ è isomorfa a $(\mathbb{Q}, \leq)$ se e solo se esiste una biiezione $h : A \to \mathbb{Q}$ tale che per ogni $a, b \in A$

$$a \preceq b \text{ se e solo se } h(a) \leq h(b).$$

2.1. Back-and-Forth e isomorfismo di ordini densi senza estremi numerabili.

La dimostrazione usa un metodo generale chiamato Back-and-Forth ("andirivieni").

Descriviamo la costruzione intuitivamente. Sia $\mathfrak{A} = (A, \preceq)$ un ordine totale denso senza estremi su un insieme numerabile.

Teorema 2.1. *Sia $\mathfrak{A} = (A, \preceq)$ una struttura con dominio A numerabile e $\preceq$ un ordine totale denso senza estremi su A. Allora esiste un isomorfismo tra $\mathfrak{A}$ e $(\mathbb{Q}, \leq)$.*

Dimostrazione. Vogliamo costruire un isomorfismo tra $\mathfrak{A}$ e $(\mathbb{Q}, \leq)$. Dato che entrambe le strutture hanno dominio numerabile, fissiamo

Fissiamo una enumerazione senza ripetizioni di A, $(a_0, a_1, a_2, a_3, \dots)$ e una enumerazione senza ripetizioni di $\mathbb{Q}$, $(b_0, b_1, b_2, b_3, \dots)$. Definiamo ricorsivamente una enumerazione $(p_1, p_2, p_3, \dots)$ senza ripetizioni e una

enumerazione $(q_1, q_2, q_3, \dots)$ senza ripetizioni in modo tale che la mappa

$$p_i \mapsto q_i$$

sia un isomorfismo tra $\mathfrak{A}$ e $\mathcal{Q}$.

Poniamo $p_0 = a_0$ e $q_0 = b_0$.

Per il passo induttivo consideriamo un n generico e assumiamo che $p_0, p_1, \dots, p_n$ e $q_0, q_1, \dots, q_n$ siano definiti.

Distinguiamo due casi.

(Caso 1) n pari. Scegliamo un elemento p_{n+1} in $A \setminus \{p_0, \dots, p_n\}$ con indice minimo in $(a_0, a_1, a_2, \dots)$. Compariamo questo elemento agli elementi già scelti. Abbiamo tre casi.

(Caso 1.1) 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 $q_{n+1} < q_m$. Questo elemento esiste perché $\mathcal{Q}$ soddisfa gli assiomi che asseriscono la non esistenza di estremi nell'ordine.

(Caso 1.2) Per ogni $m \leq n$, $p_{n+1} > p_m$. In questo caso scelgo q_{n+1} in $\mathbb{Q}$ tale che per ogni $m \leq n$ valga $q_{n+1} > q_m$. Un tale elemento esiste perché $\mathcal{Q}$ soddisfa l'assioma che esclude l'esistenza di un estremo destro nell'ordine.

(Caso 1.3) Non si danno i 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 q_{n+1} in $\mathbb{Q}$ tale che $q_{m_0} < q_{n+1} < q_{m_1}$. Un tale elemento esiste perché $\mathcal{Q}$ soddisfa l'assioma di densità.

(Caso 2) n è pari. Procediamo come nell'altro caso ma partendo da un elemento $q_{n+1} \in \mathbb{Q} \setminus \{q_0, \dots, q_n\}$.

Si mostra facilmente che la mappa definita da $p_i \mapsto q_i$ è un isomorfismo tra $\mathfrak{A}$ e $\mathcal{Q}$.

□

Rileggendo la prova precedente ci accorgiamo facilmente di aver usato soltanto le proprietà formalizzate negli enunciati della teoria DLO sopra definita, oltre all'ipotesi di numerabilità. Possiamo dunque riformulare il Teorema precedente come: ogni modello numerabile di DLO è isomorfo a $\mathcal{Q}$.

Inoltre, le sole proprietà di $\mathcal{Q}$ usate nella dimostrazione di sopra sono, oltre alla numerabilità del dominio, le proprietà espresse nella teoria DLO. Possiamo dunque ripetere la dimostrazione usando una struttura arbitraria $\mathfrak{B} = (B, \leq^{\mathfrak{B}})$ con B numerabile e tale che $\mathfrak{B} \models \mathsf{DLO}$, invece di $\mathcal{Q}$.

Otteniamo così che due modelli numerabili di DLO sono isomorfi. La stessa conclusione si ottiene tenendo fissa $\mathcal{Q}$ e facendo variare le strutture $\mathfrak{A}$ e invocando la transitività dell'isomorfismo.

Si noti che per parlare di isomorfismo tra due strutture $\mathfrak{A} = (A, \leq^{\mathfrak{A}})$ e $\mathfrak{B} = (B, \leq^{\mathfrak{B}})$ adeguate per il linguaggio $\mathcal{L} = \{\leq\}$ dobbiamo generalizzare leggermente, ma in modo del tutto naturale, la definizione di isomorfismo. Richiediamo cioè che esista una biiezione $h : A \to B$ tale che per ogni $a_1, a_2 \in A$

$$(a_1, a_2) \in \leq^{\mathfrak{A}} \text{ se e solo se } (h(a_1), h(a_2)) \in \leq^{\mathfrak{B}} .$$

Da questo isomorfismo possiamo dedurre che due modelli numerabili di DLO (ossia due ordini densi totali e senza estremi su in insieme numerabile) hanno esattamente le stesse proprietà di ordine (perché sono isomorfi): più precisamente soddisfano esattamente gli stessi enunciati nel linguaggio $\mathcal{L} = \{\leq\}$.

Resta aperta la possibilità che esista un modello non-numerabile di DLO che soddisfa enunciati diversi da $\mathcal{Q}$. Per esempio potrebbe darsi che $\mathcal{R}$ non sia elementarmente equivalente a $\mathcal{Q}$. Vedremo che non è così.

LOGICA MATEMATICA

A.A. 23/24, LOGICA PREDICATIVA, DISPENSA N. 4

SOMMARIO. Proprietà del Testimone. Funzioni di Skolem. Completezza di **DLO**. Criterio di Vaught-Tarski e Teorema di Lowenheim-Skolem.

1. Completezza di DLO

Cerchiamo di individuare un criterio sufficiente per concludere che $\mathcal{R}$ e $\mathcal{Q}$ soddisfano gli stessi enunciati.

È naturale considerare $\mathcal{Q}$ come una *sottostruttura* di $\mathcal{R}$. Per la precisione, abbiamo che $\mathbb{Q} \subseteq \mathbb{R}$ e che $\leq^{\mathcal{Q}} = \leq^{\mathcal{R}} \cap (\mathbb{Q} \times \mathbb{Q})$. In questo caso scriviamo anche $\mathcal{Q} \subseteq \mathcal{R}$.

Sia F una formula con x come unica variabile libera. È semplice osservare che se esiste un $q \in \mathbb{Q}$ tale che $\mathcal{R} \models F(x)[\binom{x}{q}]$ allora esiste un $q \in \mathbb{R}$ tale che $\mathcal{R} \models F(x)[\binom{x}{q}]$, per il semplice motivo che $\mathbb{Q} \subseteq \mathbb{R}$ e che la relazione d'ordine nei razionali è la restrizione dell'ordine sui reali. La considerazione non cambia se F ha altre variabili libere $y_1, \dots, y_n$ che vengono interpretate in $\mathbb{Q}$: Sia $F(x, y_1, \dots, y_n)$ una formula con variabili libere $x, y_1, \dots, y_n$. Sai α un assegnamento di valori in $\mathbb{Q}$ alle variabili $y_1, \dots, y_n$. Se $\mathcal{R} \models F[\binom{x, y_1, \dots, y_n}{q, \alpha(y_1), \dots, \alpha(y_n)}]$ per un $q \in \mathbb{Q}$, allora $\mathcal{R} \models F[\binom{x, y_1, \dots, y_n}{q, \alpha(y_1), \dots, \alpha(y_n)}]$ (Esercizio, cfr. la dimostrazione della Proposizione 1.1 più avanti). Da questo segue che, se α è un assegnamento in $\mathbb{Q}$ e $\mathcal{Q} \models \exists x F[\alpha]$ allora $\mathcal{R} \models \exists x F[\alpha]$.

Non è detto in generale che valga il contrario, ossia la seguente proprietà:

Proprietà del Testimone tra $\mathcal{R}$ e $\mathcal{Q}$
Per ogni formula F con x tra le sue variabili libere, **per ogni assegnamento** α **in** $\mathbb{Q}$, se

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

allora **esiste un** $q \in \mathbb{Q}$ tale che

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

La Proprietà del Testimone si rivela essere una condizione sufficiente per dedurre che $\mathcal{Q}$ e $\mathcal{R}$ soddisfano gli stessi enunciati (in questo caso diciamo che sono *elementarmente equivalenti*, e scriviamo $\mathcal{Q} \equiv \mathcal{R}$).

Come al solito, conviene dimostrare la proprietà analoga relativa a formule:

Proposizione 1.1. *Se vale la Proprietà del Testimone tra $\mathcal{Q}$ e $\mathcal{R}$ allora per ogni F e per ogni α su $\mathbb{Q}$,*

$$\mathcal{Q} \models F[\alpha] \text{ se e solo se } \mathcal{R} \models F[\alpha].$$

Dimostrazione. Per induzione su F.

Se F è atomica e di forma $(x \leq y)$, e abbiamo

$$\mathcal{Q} \models (x \leq y)[\alpha] \text{ sse } \alpha(x) \leq^{\mathcal{Q}} \alpha(y) \text{ sse } \alpha(x) \leq^{\mathcal{R}} \alpha(y) \text{ sse } \mathcal{R} \models (x \leq y)[\alpha].$$

Se F è atomica e di forma $(x = y)$ l'argomento è completamente analogo.

Il caso dei Booleani è semplice (Esercizio).

Il caso dei quantificatori segue dall'ipotesi: supponiamo $\mathcal{Q} \models \exists x F(x)[\alpha]$. Per definizione vale sse esiste $q \in \mathbb{Q}$ tale che $\mathcal{Q} \models F(x)[\alpha \binom{x}{q}]$. Per ipotesi induttiva questo implica che $\mathcal{R} \models F(x)[\alpha \binom{x}{q}]$. Per definizione di $\models$ e dato che $\mathbb{Q} \subseteq \mathbb{R}$ questo implica che $\mathcal{R} \models \exists x F(x)[\alpha]$.

Per l'altro verso supponiamo che $\mathcal{R} \models \exists x F(x)[\alpha]$, per α assegnamento in $\mathbb{Q}$. Per definizione vale sse esiste $r \in \mathbb{R}$ tale che $\mathcal{R} \models F(x)[\alpha \binom{x}{r}]$. Per la Proprietà dei Testimoni questo implica che esiste $q \in \mathbb{Q}$ tale che

Appunti preparati da Lorenzo Carlucci, lorenzo.carlucci@uniroma1.it.

$\mathcal{R} \models F(x)[\alpha\binom{x}{q}]$. Per ipotesi induttiva su F questo vale sse esiste $q \in \mathbb{Q}$ tale che $\mathcal{Q} \models F(x)[\alpha\binom{x}{q}]$, e questo per definizione di soddisfacibilità equivale a $\mathcal{Q} \models \exists x F(x)[\alpha]$. □

Della Proposizione precedente ci interessa particolarmente il seguente Corollario immediato.

Corollario 1.2. *Se vale la Proprietà del Testimone tra $\mathcal{Q}$ e $\mathcal{R}$, allora, per ogni enunciato E (nel linguaggio degli ordini) vale:*

$$\mathcal{Q} \models E \text{ sse } \mathcal{R} \models E,$$

ossia: $\mathcal{Q}$ e $\mathcal{R}$ sono elementarmente equivalenti.

Non dimostreremo che la Proprietà del Testimone vale tra $\mathcal{R}$ e $\mathcal{Q}$ bensì tra $\mathcal{R}$ e una struttura $\mathcal{A}$ numerabile che è sottostruttura di $\mathcal{A}$ e di cui $\mathcal{Q}$ è sottostruttura. Come vedremo questo sarà sufficiente a concludere che $\mathcal{R} \equiv \mathcal{Q}$ e, con una semplice generalizzazione, che per ogni modello $\mathcal{B}$ non numerabile di **DLO**, vale $\mathcal{Q} \equiv \mathcal{B}$.

Si invita il lettore a verificare che la Proposizione precedente vale nella forma seguente (dove $\mathcal{A}$ sostituisce $\mathcal{Q}'$). La Proprietà del Testimone tra $\mathcal{A}$ e $\mathcal{R}$ è definita come sopra, con $\mathcal{A}$ al posto di $\mathcal{Q}$, ossia:

Proprietà del Testimone tra $\mathcal{R}$ e $\mathcal{A}$ sottostruttura di $\mathcal{R}$
Per ogni formula F con x tra le sue variabili libere, **per ogni assegnamento** α **in** $\mathcal{A}$, se

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

allora **esiste un** $a \in A$ tale che

$$\mathcal{R} \models F(x)[\alpha\binom{x}{a}].$$

Proposizione 1.3. *Sia $\mathcal{A} = (A, \leq^{\mathcal{A}})$ una sottostruttura di $\mathcal{R}$. Se vale la Proprietà del Testimone tra $\mathcal{A}$ e $\mathcal{R}$ allora per ogni F e per ogni α su $\mathbb{Q}$,*

$$\mathcal{A} \models F[\alpha] \text{ se e solo se } \mathcal{R} \models F[\alpha].$$

La dimostrazione è identica alla precedente. Vedremo più avanti come generalizzare ulteriormente la Proposizione usando una struttura $\mathcal{B}$ arbitraria invece di $\mathcal{R}$.

Specificando al caso di enunciati la Proposizione precedente otteniamo il seguente Corollario immediato.

Corollario 1.4. *Se $\mathcal{A}$ è una sottostruttura di $\mathcal{R}$ e vale la proprietà del Testimone tra $\mathcal{A}$ e $\mathcal{R}$ allora, per ogni eunciato E (nel linguaggio degli ordini) vale:*

$$\mathcal{A} \models E \text{ sse } \mathcal{R} \models E,$$

ossia: $\mathcal{A}$ e $\mathcal{R}$ sono elementarmente equivalenti.

Mostriamo come, partendo da $\mathcal{Q}$ è possibile ottenere una sottostruttura numerabile $\mathcal{A}$ di $\mathcal{R}$ per cui vale la Proprietà del Testimone relativamente a $\mathcal{R}$.

Sia F una formula del linguaggio di **DLO** e siano $x, y_1, \ldots, y_n$ tutte le sue variabili libere.

Se $\mathcal{R} \models \exists x F(x)[\binom{y_1,\ldots,y_n}{b_1,\ldots,b_n}]$ per $b_1, \ldots, b_n \in \mathbb{R}$, allora, per definizione di soddisfacibilità, esiste un $b \in \mathbb{R}$ tale che $\mathcal{R} \models F[\binom{x,y_1,\ldots,y_n}{b,b_1,\ldots,b_n}]$.

Chiamiamo *funzione di Skolem*[1] della formula F relativamente alla variabile x la funzione $f_{F,x} : \mathbb{R}^n \to \mathbb{R}$ che associa a ogni scelta di $(b_1, \ldots, b_n) \in \mathbb{R}^n$ un tale $b \in \mathbb{R}$ in modo canonico (usiamo implicitamente l'Assioma della Scelta).

Consideriamo l'insieme $\mathcal{S}$ di tutte queste funzioni, al variare di F tra le formule e di x tra le variabili del linguaggio.

Procediamo come segue. Partiamo ponendo $A_1 = \mathbb{Q}$. Definiamo A_2 chiudendo $\mathbb{Q}$ sotto tutte le funzioni di Skolem valiutate su elementi di A_1, ossia

$$A_2 = \{f(q_1, \ldots, q_n) \,:\, (q_1, \ldots, q_n) \in \mathbb{Q}; f \in \mathcal{S}\}.$$

[1]Dal nome del logico norvegese Thoralf Skolem che ha ideato questo metodo a inizio Novecento.

Analogamente definiamo A_3 come la chiusura di A_2 sotto tutte le funzioni di Skolem valutate su elementi in A_2, ossia

$$A_3 = \{f(q_1, \ldots, q_n) \,:\, (q_1, \ldots, q_n) \in A_2; f \in \mathcal{S}\}.$$

e così via. In generale poniamo

$$A_{k+1} = A_k \cup \{f(q_1, \ldots, q_n) \,:\, (q_1, \ldots, q_n) \in A_k; f \in \mathcal{S}\}.$$

Definiamo

$$A = \bigcup_{k \in \mathbb{N}} A_k.$$

Si osserva facilmente che A è chiuso sotto funzioni di Skolem: se $a_1, \ldots, a_n \in A$ e $f \in \mathcal{S}$, allora $f(a_1, \ldots, a_n) \in A$. Esiste infatti un k tale che $a_1, \ldots, a_n \in A_k$. Ma allora $f(a_1, \ldots, a_n) \in A_{k+1}$.

Consideriamo la struttura $\mathcal{A} = (A, \leq^{\mathcal{A}})$ dove $\leq^{\mathcal{A}}$ è definito come $\leq^{\mathcal{R}} \cap A \times A$.

Verifichiamo che $\mathcal{A}$ soddisfa la Proprietà del Testimone relativamente a $\mathcal{R}$, ossia: per ogni formula F, per ogni α assegnamento in A, se

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

allora esiste $a \in A$ tale che

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

Basta considerare la funzione di Skolem $f_{F,x}$ associata a F, relativamente alla variabile x. Se $x, y_1, \ldots, y_n$ sono tutte le variabili libere di F ho che

$$\mathcal{R} \models F[b, \alpha(y_1), \ldots, \alpha(y_n)]$$

dove $b = f_{F,x}(\alpha(y_1), \ldots, \alpha(y_n))$ (si noti che gli $\alpha(y_i)$ sono in A per scelta di α). Inoltre $b \in A$ perché A è chiuso sotto funzioni di Skolem!

Dunque possiamo concludere, in base alla proposizione precedente, che $\mathcal{A} \equiv \mathcal{R}$.

Si verifica inoltre facilmente che la cardinalità di A è numerabile. Il numero delle funzioni di Skolem è numerabile: abbiamo una funzione per ogni scelta di una formula del linguaggio e di una variabile libera; si tratta dunque di un prodotto di due insiemi numerabili. Si dimostra facilmente per induzione che ogni A_k è numerabile. Per $A_1 = \mathbb{Q}$ è ovvio. Per A_{k+1}: si tratta dell'unione di A_k (numerabile) con la sua chiusura sotto funzioni di Skolem. Queste sono in quantità numerabile come osservato sopra. La chiusura di A_k sotto funzioni di Skolem è l'unione delle immagini di tipo $f[A_k^n]$ al variare di f tra le funzioni di Skolem. Ogni insieme di tipo $f[A_k^n]$ è numerabile, perché A_k è numerabile (per ipotesi induttiva). Dunque la chiusura di A_k sotto funzioni di Skolem è una unione numerabile di insiemi nuemrabile; ed è dunque numerabile! Dunque l'intero A_{k+1} è numerabile.

Abbiamo dimostrato la seguente proposizione.

Proposizione 1.5. *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.*

Si osserva che $\mathcal{A}$ è anche un modello numerabile di **DLO**, perché $\mathcal{R} \models$ **DLO**. Dunque $\mathcal{A}$ è isomorfo a $\mathcal{Q}$. Dunque abbiamo $\mathcal{Q} \equiv \mathcal{R}$: le due strutture soddisfano gli stessi enunciati nel linguaggio degli ordini!

Corollario 1.6. *$\mathcal{Q}$ e $\mathcal{R}$ soddisfano gli stessi enunciati nel linguaggio degli ordini, i.e.,*

$$\mathcal{Q} \equiv \mathcal{R}.$$

Dimostrazione. Abbiamo visto come costruire $\mathcal{A}$ sottostruttura numerabile di $\mathcal{A}$ tale che $\mathcal{A} \equiv \mathcal{R}$. In particolare, dato che $\mathcal{R} \models$ **DLO**, abbiamo $\mathcal{A} \models$ **DLO**. Dunque $\mathcal{A}$ è un modello numerabile di **DLO**. Dunque è isomorfa a $\mathcal{Q}$, e dunque $\mathcal{Q} \equiv \mathcal{A}$. Si conclude che $\mathcal{Q} \equiv \mathcal{R}$. □

Il metodo appena descritto è completamente generale. Come prima osservazione si ripercorrano le dimostrazioni precedenti partendo non da $\mathcal{R}$ bensì da una qualunque struttura $\mathcal{B}$ modello non-numerabile di **DLO** (invece che da $\mathcal{R}$).

La Proprietà del Testimone si esprime adesso relativamente non a $\mathcal{Q}$ ma a una arbitraria sottostruttura $\mathcal{A}$ di $\mathcal{B}$, ossia una strtuttra di tipo $(A, <^{\mathcal{A}})$ con $A \subseteq \mathcal{B}$ e $<^{\mathcal{A}} = <^{\mathcal{B}} \cap (A \times A)$. La dimostrazione che la proprietà del testimone implica l'equivalenza elementare si trasferisce senza modifiche a questo setting: Se $\mathcal{A}$ è una

sottostruttura di $\mathcal{B}$ che soddisfa la Proprietà del Testimone, allora $\mathcal{A}$ e $\mathcal{B}$ soddisfano gli stessi enunciati. Si ripercorra la dimostrazione della Proposizione 1.3.

La costruzione di una sottostruttura equivalente a $\mathcal{R}$ ottenuta come chiusura di $\mathbb{Q}$ sotto le funzioni di Skolem si traferisce verbatim a questo setting: Si parta da una struttura $\mathcal{B} \models \mathbf{DLO}$ e da un suo arbitrario sottinsieme X numerabile. Ripercorrendo la costruzione nella dimostrazione della Proposizione 1.5, otteniamo una struttura numerabile $\mathcal{A}$ sottostruttura di $\mathcal{B}$ ed elementarmente equivalente a $\mathcal{B}$ come chiusura di X sotto funzioni di Skolem. Come osservato sopra, $\mathcal{A} \models \mathbf{DLO}$ ed è dunque isomorfa ed equivalente a $\mathcal{Q}$.

Da queste considerazioni segue che tutti modelli di **DLO**, siano essi numerabili o meno, soddisfano gli stessi enunciati, i.e., esattamente gli stessi enunciati di $\mathcal{Q}$.

Corollario 1.7. *La teoria* **DLO** *è completa.*

Dimostrazione. Un modello arbitrario di **DLO** soddisfa esattamente gli stessi enunciati di $\mathcal{Q}$. Dunque gli enunciati veri in tutti e soli i modelli di **DLO** (i.e. $Cons(\mathbf{DLO})$) sono esattamente gli enunciati veri nella singola struttura $\mathcal{Q}$. Quest'ultimo insieme è una teoria completa, come già osservato. □

2. Extra: Criterio di Vaught-Tarski e Teorema di Lowenheim-Skolem

Possiamo generalizzare ulteriormente i risultati di sopra con modifiche minime alle dimostrazioni date.

In primo luogo generalizziamo il concetto di sottostruttura a linguaggi arbitrari.

Definizione 2.1 (Sottostruttura)**.** Siano $\mathcal{A}$ e $\mathcal{B}$ due strutture per lo stesso linguaggio $\mathcal{L}$. Si dice che $\mathcal{A}$ è *sottostruttura* di $\mathcal{B}$ se e solo se

(1) $A \subseteq B$
(2) Per ogni simbolo di relazione R a n argomenti, $R^{\mathcal{A}} = R^{\mathcal{B}} \cap (A^n)$.
(3) Per ogni simbolo di costante c, $c^{\mathcal{A}} = c^{\mathcal{B}}$.
(4) Per ogni simbolo di funzione f a n argomenti, $f^{\mathcal{A}} = f^{\mathcal{B}}|A^n$.

Rivisitando le dimostrazioni della sezione precedente otteniamo i seguenti risultati generali. Innanzitutto formalizziamo la Proprietà del Testimone per due strutture $\mathcal{A}$ e $\mathcal{B}$ con $\mathcal{A}$ sottostruttura di $\mathcal{B}$.

Proprietà del Testimone tra $\mathcal{A}$ e $\mathcal{B}$
Per ogni formula F con x tra le sue variabili libere, **per ogni assegnamento** α **in** A, se

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

allora **esiste un** $a \in A$ tale che

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

Teorema 2.2. *Se $\mathcal{A}$ è una sottostruttura di $\mathcal{B}$ che soddisfa la Proprietà del Testimone rispetto a $\mathcal{B}$ allora $\mathcal{A} \equiv \mathcal{B}$ ($\mathcal{A}$ e $\mathcal{B}$ soddisfano gli stessi enunciati).*

Dimostrazione. La dimostrazione si ripete quasi verbatim, nella sua formulazione per formule: Se $\mathcal{A}$ è sottostruttura di $\mathcal{B}$ e soddisfa la Proprietà del Testimone, allora, per ogni formula F e ogni assegnamento α in A, vale

$$\mathcal{A} \models F[\alpha] \text{ se e solo se } \mathcal{B} \models F[\alpha].$$

L'unica differenza nella dimostrazione riguarda il caso atomico.

Supponiamo che $\mathcal{A} \models R(t_1, \ldots, t_n)[\alpha]$, con α un assegnamento in A. Questo vale sse $(\alpha(t_1), \ldots, \alpha(t_n)) \in R^{\mathcal{A}}$. Ma $R^{\mathcal{A}} = R^{\mathcal{B}} \cap (A^n)$, e possiamo concludere che l'appartenenza precedente vale sse $(\alpha(t_1), \ldots, \alpha(t_n)) \in R^{\mathcal{B}}$, che vale sse $\mathcal{B} \models R(t_1, \ldots, t_n)[\alpha]$, per definizione di soddisfazione.

(NB: L'argomento precedente contiene un piccolissimo gap: quale? E come colmarlo?).

Il resto della dimostrazione si ripete verbatim. □

Generalizzando la Proposizione 1.5 otteniamo il secondo Teorema.

Teorema 2.3 (Teorema di Lowenheim-Skolem)**.** *Sia $\mathcal{B}$ una struttura infinita adeguata per un linguaggio $\mathcal{L}$. Sia $X \subseteq B$ un sottinsieme del suo dominio. Esiste una struttura $\mathcal{A}$ adeguata per $\mathcal{L}$ tale che*

(1) $X \subseteq A$
(2) $\mathcal{A}$ *è una sottostruttura di* $\mathcal{B}$
(3) $\mathcal{A}$ *soddisfa la Proprietà del Testimone rispetto a* $\mathcal{B}$.
(4) *Se il linguaggio e l'insieme* X *sono numerabili, allora* A *è numerabile.*

Dimostrazione. La dimostrazione per $\mathcal{Q}$ e $\mathcal{R}$ si trasferisce quasi verbatim. La costruzione dell'insieme A come chiusura sotto funzioni di Skolem dell'insieme X (invece di $\mathbb{Q}$) si ripete verbatim. A questo punto, per procedere, è necessario dimostrare che possiamo identificare una struttura $\mathcal{A}$ con dominio A. Per definire le relazioni in $\mathcal{A}$, basta generalizzare quanto visto per $\mathcal{Q}$ e $\mathcal{R}$: definiamo $R^{\mathcal{A}}$ come $R^{\mathcal{B}} \cap (A^n)$, per ogni simbolo di relazione a n posti nel linguaggio. Dobbiamo però assegnare una interpretazione alle costanti e alle funzioni in modo tale da ottenere una sottostruttura. Dobbiamo quindi designare un $c^{\mathcal{A}}$ in A per ogni simbolo di costante c e, per ogni simbolo di funzione f a n posti, una interpretazione $f^{\mathcal{A}}$; in modo tale da ottenere una sottostruttura di $\mathcal{B}$. Quindi non abbiamo molta scelta: dobbiamo avere $c^{\mathcal{A}} = c^{\mathcal{B}}$ e $f^{\mathcal{A}} = f^{\mathcal{B}}|A^n$. Dobbiamo dunque soltanto assicurarci che questi oggetti siano già in A! Questo segue come caso particolare dalla chiusura di A sotto funzioni di Skolem. Vediamo i dettagli. Sia c un simbolo di costante. Consideriamo la formula $(x = c)$. Ovviamente $\mathcal{B} \models \exists x(x = c)$. Sia $b \in B$ tale che $\mathcal{B} \models (x = c)[\binom{x}{b}]$. Necessariamente vale $b = c^{\mathcal{B}}$, perché il valore delle costanti è fissato dalla struttura. D'altra parte l'elemento b è necessariamente il valore della funzione di Skolem associata alla formula $(x = c)$ relativamente alla variabile x (abbiamo denotato questa funzione con $f_{(x=c),x}$), per qualunque argomento scelto in B. In particolare, per $a \in A_1 \subseteq A$, vale $f_{(x=c),x}(a) = b$. Per costruzione abbiamo $b \in A_2$ e dunque $b \in A$. Questa è la conclusione desiderata: $c^{\mathcal{B}} \in A$.

Passiamo ai termini funzionali. Sia g un simbolo di funzione a n posti nel linguaggio. Consideriamo la formula $g(y_1, \ldots, y_n) = x$. Ovviamente vale, per ogni $b_1, \ldots, b_m \in B$:

$$\mathcal{B} \models \exists x(g(y_1, \ldots, y_n) = x)[\binom{y_1, \ldots, y_m}{b_1, \ldots, b_m}],$$

perché le funzioni in $\mathcal{B}$ sono ovunque definite. In particolare vale per $a_1, \ldots, a_m \in A \subseteq B$. Consideriamo la funzione di Skolem associata alla formula $g(y_1, \ldots, y_n) = x$, relativamente alla variabile x; ossia $f_{(g(y_1,\ldots,y_n)=x),x}$. Per ogni $a_1, \ldots, a_m \in A$, se $f_{(g(y_1,\ldots,y_n)=x),x}(a_1, \ldots, a_m) = b \in B$ allora vale

$$\mathcal{B} \models g(y_1, \ldots, y_n) = x)[\binom{y_1, \ldots, y_m, x}{a_1, \ldots, a_m, b}].$$

Questo è possibile se e soltanto se $f^{\mathcal{B}}(a_1, \ldots, a_m) = b$. Questo valore è in A per chiusura sotto le funzioni di Skolem.

La cardinalità di A è valutata come nel caso della chiusura di Skolem di $\mathbb{Q}$: si tratta di una unione numerabile di insiemi numerabili, se il linguaggio (e dunque l'insieme delle funzioni di Skolem) è numerabile.

Questo conclude l'argomento. □

Corollario 2.4. *Se una teoria* T *in un linguaggio numerabile ha un modello infinito allora ha un modello numerabile.*

Dimostrazione. Sia $\mathcal{B} \models T$ più che numerabile. Sia $X \subseteq B$ un sottinsieme numerabile del suo dominio. Per il Teorema precedente esiste una struttura numerabile $\mathcal{A}$ sottostruttura di $\mathcal{B}$ che soddisfa la Proprietà del Testimone rispetto a $\mathcal{B}$. Per il Criterio di Tarski-Vaught questo implica che $\mathcal{A}$ e $\mathcal{B}$ soddisfano gli stessi enunciati. Dunque $\mathcal{A} \models T$. □

Osservazione 2.5. Il Teorema precedente dà luogo a un interessante paradosso che abbiamo già dedotto dalla nostra dimostrazione del Teorema di Completezza. Consideriamo la teoria formale degli insiemi. Una tale teoria è esprimibile in un linguaggio predicativo con un unico simbolo di relazione binaria, $\in$, per denotare l'appartenenza insiemistica. La teoria classica è quella di Zermelo-Frankel, cui si può aggiungere l'Assioma della Scelta. Come noto, in una teoria del genere è possibile dimostrare l'esistenza di insiemi non-numerabili; per esempio $\mathbb{R}$. Abbiamo la situazione seguente: da T segue l'enunciato formale corrispondente a "Esiste un insieme non-numerabile", ma, se T ha un qualche modello, allora per il Teorema di Lowenheim-Skolem, T ha anche un modello numerabile. Esiste dunque una struttura numerabile in cui è soddisfatto l'enunciato "Esiste un insieme non-numerabile". Questo è noto come "Paradosso di Skolem".

LOGICA MATEMATICA

A.A. 23/24, LOGICA PREDICATIVA, DISPENSA N. 5

SOMMARIO. Isomorfismi di strutture, equivalenza elementare.

In questa dispensa raccogliamo le formulazioni generali del concetto di isomorfismo di strutture e i dettagli delle dimostrazioni che l'isomorfismo implica l'equivalenza elementare.

1. Isomorfismi di strutture

In Matematica è usuale identificare due oggetti (gruppi, grafi, anelli, etc.) modulo un isomorfismo. Due gruppi isomorfi sono "moralmente" lo stesso gruppo, in particolare nel senso che non sarà possibile individuare/formulare una proprietà che li distingua. I due gruppi isomorfi risulteranno indistinguibili o indiscernibili da proprietà che possiamo esprimere nel linguaggio dei gruppi. La loro differenza consiste unicamente nel fatto che i loro elementi sono oggetti di tipo diverso e che le operazioni definite su di essi sono differenti, per quanto si comportino in tutto e per tutto nello stesso modo.

Introduciamo una definizione di isomorfismo adatta a strutture arbitrarie. Si tratta di una mera generalizzazione della nozione abituale in Algebra.

Definizione 1.1 (Isomorfismo)**.** Siano $\mathcal{A}, \mathcal{B}$ strutture per il linguaggio $\mathcal{L}$. $\mathcal{A}$ è isomorfa $\mathcal{B}$ (scriviamo $\mathcal{A} \simeq \mathcal{B}$) se e solo se esiste una mappa $h : A \to B$ 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.

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

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

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

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

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

(Diciamo che h commuta con il simbolo di funzione f).

La Proposizione seguente mostra che l'isomorfismo è una condizione sufficiente affinché due strutture sia indistinguibili per mezzo di enunciati.

Proposizione 1.2. *Se $\mathcal{A}$ e $\mathcal{B}$ sono strutture per il linguaggio $\mathcal{L}$, ed esiste un isomorfismo tra $\mathcal{A}$ e $\mathcal{B}$, allora, per ogni enunciato E nel linguaggio, vale*

$$\mathcal{A} \models E \text{ se e solo se } \mathcal{B} \models E.$$

Dimostrazione. Conviene considerare una versione per formule e non solo per enunciati. Per ogni formula $F(x_1, \dots, x_n)$, per ogni $(a_1, \dots, a_n) \in A^n$,

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

Questa formulazione può essere dimostrata facilmente per induzione sulla complessità della formula.

Il primo caso è una formula atomica $(t_1 = t_2)$. Questa formula può contenere variabili libere (che compaiono nei termini), siano esse comprese in $\{x_1, \dots, x_n\}$. In analogia alla notazione usata per le formule

Appunti preparati da Lorenzo Carlucci, lorenzo.carlucci@uniroma1.it.

scriviamo $t(x_1, \ldots, x_n)$ per indicare che le variabili (libere) del termine t sono comprese in $\{x_1, \ldots, x_n\}$. In tal caso, se α è un assegnamento e $\alpha(x_i) = a_i$, denotiamo $\alpha(t)$ con $t[a_1, \ldots, a_n]$.

La nostra tesi è che per ogni α

$$\mathcal{A} \models (t_1 = t_2)[\alpha]$$

vale se e solo se

$$\mathcal{B} \models (t_1 = t_2)[h(\alpha)],$$

dove con $h(\alpha)$ indichiamo l'assegnamento in B che associa alla variabile v_i il valore $h(\alpha(v_i))$.

$$\mathcal{A} \models (t_1 = t_2)[\alpha]$$

vale per definizione se e solo se $\alpha(t_1)$ è identico ad $\alpha(t_2)$. Siano $a_1, \ldots, a_n in A$ tali che $\alpha(x_i) = a_i$. Possiamo riscrivere la condizione precedente come segue:

$$t_1[a_1, \ldots, a_n] = t_2[a_1, \ldots, a_n].$$

D'altra parte la condizione

$$\mathcal{B} \models (t_1 = t_2)[h(\alpha)],$$

equivale per definizione a

$$t_1[h(a_1), \ldots, h(a_n)] = t_2[h(a_1), \ldots, h(a_n)].$$

Dato che h è una biiezione, da

$$t_1[a_1, \ldots, a_n] = t_2[a_1, \ldots, a_n]$$

segue

$$h(t_1[a_1, \ldots, a_n]) = h(t_2[a_1, \ldots, a_n]).$$

Per colmare il divario basta dimostrare che

$$h(t_1[a_1, \ldots, a_n]) = t_1[h(a_1), \ldots, h(a_n)],$$

e

$$h(t_2[a_1, \ldots, a_n]) = t_2[h(a_1), \ldots, h(a_n)],$$

ossia che l'isomorfismo h commuta con le interpretazioni dei termini. In generale conviene dimostrare per induzione sulla complessità dei termini, il seguente Lemma.

Lemma 1.3. *Sia t un termine con variabili in $\{x_1, \ldots, x_n\}$. Per ogni $(a_1, \ldots, a_n) \in A^n$:*

$$h(t[a_1, \ldots, a_n] = t[h(a_1), \ldots, h(a_n)].$$

Consideriamo il caso di una formula atomica $R(t_1, \ldots, t_k)$, con variabili in $\{x_1, \ldots, x_n\}$. In questo caso la tesi è che per ogni $(a_1, \ldots, a_n) \in A^n$:

$$\mathcal{A} \models R(t_1, \ldots, t_k)[a_1, \ldots, a_n] \text{ sse } \mathcal{B} \models R(t_1, \ldots, t_k)[h(a_1), \ldots, h(a_n)].$$

Questo segue immediatamente dalla Definizione di isomorfismo e dal Lemma sui termini.

I casi booleani sono ovvii.

Sia ora F una formula $\exists x F(x)$, con variabili libere in $\{x_1, \ldots, x_n\}$. In questo caso la tesi è che

$$\mathcal{A} \models \exists x F(x)[a_1, \ldots, a_n] \text{ sse } \mathcal{B} \models \exists x F(x)[h(a_1), \ldots, h(a_n)].$$

La condizione

$$\mathcal{A} \models \exists x F(x)[a_1, \ldots, a_n]$$

equivale per definizione a

$$\text{Esiste } a \in A \;\; \mathcal{A} \models F[\begin{pmatrix} x, x_1, \ldots, x_n \\ a, a_1, \ldots, a_n \end{pmatrix}]$$

Per ipotesi induttiva la condizione

$$\mathcal{A} \models F[\begin{pmatrix} x, x_1, \ldots, x_n \\ a, a_1, \ldots, a_n \end{pmatrix}]$$

equivale a

$$\mathcal{B} \models F[\begin{pmatrix} x, x_1, \ldots, x_n \\ h(a), h(a_1), \ldots, h(a_n) \end{pmatrix}].$$

Per concludere basta osservare che

$$\text{Esiste } a \in A \;\; \mathcal{B} \models F[\binom{x, x_1, \ldots, x_n}{h(a), h(a_1), \ldots, h(a_n)}]$$

vale se e solo se

$$\text{Esiste } b \in B \;\; \mathcal{B} \models F[\binom{x, x_1, \ldots, x_n}{b, h(a_1), \ldots, h(a_n)}].$$

In un verso è una semplice conclusione *a fortiori* (Esiste $a \in A$ tale che $h(a)$... implica esiste $b \in B$ — ossia $h(a)$ — tale che b...). Nell'altro verso segue dalla suriettività di h: se esiste un $b \in B$ tale che b... allora esiste una sua preimmagine $a \in A$ e dunque esiste un $a \in A$ tale che $h(a)$... □

Consideriamo ora il caso di $\mathcal{A}$ e $\mathcal{B}$ finiti, osservando che in questo caso soddisfare gli stessi enunciati implica l'isomorfismo tra le strutture.

Proposizione 1.4. *Se $\mathcal{A}$ e $\mathcal{B}$ sono finite e soddisfano gli stessi enunciati allora sono isomorfe.*

Dimostrazione. L'idea è semplicemente di descrivere completamente la struttura $\mathcal{A}$ con un enunciato $E_{\mathcal{A}}$, così che ogni altra struttura $\mathcal{B}$ che soddisfi $E_{\mathcal{A}}$ sia necessariamente isomorfa a $\mathcal{A}$.

Per semplicità consideriamo $\mathcal{A}$ per un linguaggio con un singolo simbolo di relazione binaria R. Dunque $\mathcal{A} = (\{a_1, a_2, \ldots, a_n), R^{\mathcal{A}})$, con $R^{\mathcal{A}} \subseteq A \times A$.

Si definisca un enunciato $E_{\mathcal{A}}$ nelle variabili $\{v_1, \ldots, v_n\}$ che descriva completamente la struttura $\mathcal{A}$.

□

LOGICA MATEMATICA

A.A. 23/24, LOGICA PREDICATIVA, DISPENSA N. 6

Sommario. Calcolo dei Predicati.

1. Calcolo dei Predicati

Definiamo una nozione di dimostrazione logica formale (per il I ordine). Una dimostrazione sarà una sequenza finita di formule, dove ogni formula è o un assioma (logico), o una assunzione (non logica), o è ottenuta da una o più formule precedenti per applicazione di una regola di inferenza. Il calcolo che definiamo ha la caratteristica che gli assiomi sono algoritmicamente riconoscibili e che l'applicazione corretta di una regola di inferenza è algoritmicamente testabile.

Gli assiomi sono fatti in modo da catturare tutte e sole le formule logicamente valide. Dimostreremo, con il Teorema di Completezza, che è proprio così: una formula è dimostrabile se e solo se è valida. In questo modo avremmo sostituito una nozione semantica e infinitaria (quella di validità in tutti i modelli) con una nozione sintattica e finitaria (quella di dimostrabilità). Una conseguenza di rilievo è che l'insieme delle formule valide è algoritmicamente enumerabile.

Il particolare tipo formalismo che ora introduciamo è detto alla Hilbert ed è basato su assiomi e regole di deduzione. Non ci affezioniamo al formalismo (ne vedremo altri e useremo di volta in volta il più utile ce ne sono molti e tutti effettivamente equivalenti). Ci concentriamo sulle proprietà essenziali che ci serviranno nel seguito. Non perdiamo tempo a svolgere derivazioni formali come esercizio (le svolgeremo quando serviranno per un teorema). Ci limitiamo ad alcune proprietà fondamentali. Gli assiomi del Calcolo dei Predicati sono i seguenti (dove F e G sono formule arbitrarie). Il primo gruppo sono Assiomi Proposizionali, il secondo gruppo sono Assiomi Predicativi. Un terzo gruppo, gli Assiomi dell'Identità, viene aggiunto quando si sceglie di trattare la relazione di uguaglianza $=$ come un simbolo logico speciale, e si richiede nella definizione di soddisfacibilità che questo simbolo venga interpretato come l'uguaglianza tra elementi del modello. Un approccio alternativo è indicato nella prossima sezione.

(Schemi di) Assiomi Proposizionali.

(1) $F \to (G \to F)$
(2) $(F \to (G \to H)) \to ((F \to G) \to (F \to H))$
(3) $(\neg F \to \neg G) \to ((\neg F \to G) \to F)$
(4) $(F \wedge G) \to F$
(5) $(F \wedge G) \to G$
(6) $(H \to F) \to ((H \to G) \to (H \to (F \wedge G)))$
(7) $F \to (F \vee G)$
(8) $G \to (F \vee G)$
(9) $(F \to H) \to ((G \to H) \to ((F \vee G) \to H))$

(Schemi di) Assiomi Predicativi.

(10) $\forall x F \to F[x/t]$, con t termine libero per x in F.
(11) $\forall x(F \to G) \to (F \to (\forall x)G)$, con F senza occorrenze libere di x.

Appunti preparati da Lorenzo Carlucci, lorenzo.carlucci@uniroma1.it.

Si intende che ogni formula ottenuta dagli schemi precedenti sostituendo coerentemente alle variabili F, G, H delle formule del linguaggio predicativo è un assioma del Calcolo dei Predicati.

Gli assiomi di sopra non hanno clausole per $\exists$. Questa è soltanto una scelta di economia, perché si può definire $\exists$ come $\neg\forall\neg$. Altrimenti, si può introdurre come assioma la seguente doppia implicazione

$$\exists x F \leftrightarrow \neg\forall x \neg F.$$

Giustifichiamo le restrizioni agli Assiomi 10 e 11.

Se t non è libero per x in $F(x)$ nell'Assioma 10, si può andare incontro a problemi. Sia $F(v_1)$ la formula $(\exists v_2)\neg R(v_1, v_2)$. Sia t la variabile v_2. t non è libero per v_1 in $F(v_1)$. Se applicassimo l'Assioma 10, avremmo

$$(\forall v_1)(\exists v_2)\neg R(v_1, v_2) \rightarrow \exists v_2 \neg R(v_2, v_2).$$

Se interpretiamo l'implicazione precedente in una struttura con almeno due elementi dove R è interpretata come il l'identità, verifichiamo l'antecedente ma non il conseguente.

Siano F e G entrambe identiche alla formula $R(v_1)$. Ovviamente v_1 è libera in F. Consideriamo l'applicazione dell'Assioma (11)

$$(\forall v_1)(R(v_1) \rightarrow R(v_1)) \rightarrow (R(v_1) \rightarrow (\forall v_1)R(v_1)).$$

Interpretiamo la formula nella struttura che ha per dominio $\mathbf{N}$ e che interpreta R come l'insieme dei numeri pari. L'enunciato $(\forall v_1)R(v_1)$ non è soddisfatto da nessun assegnamento. Tutti gli assegnamenti che mandano v_1 in un numero pari soddisfano sia l'antecedente $(\forall v_1)(R(v_1) \rightarrow R(v_1))$ (che è logicamente valido) che la premessa del conseguente, $R(v_1)$.

Se scegliamo di avere $=$ come simbolo speciale nel linguaggio, aggiungiamo al calcolo i seguenti assiomi. In questo caso parliamo di *Calcolo dei Predicati con uguaglianza*.

Assiomi dell'uguaglianza

(10) Per ogni simbolo di funzione f, il seguente enunciato è un assioma.

$$\forall x_1 \ldots \forall x_n \forall y_1 \ldots \forall y_n (x_1 = y_1 \wedge \cdots \wedge x_n = y_n \rightarrow (f(x_1, \ldots, x_n) = f(y_1, \ldots, y_n)),$$

(11) Per ogni simbolo di relazione R, il seguente enunciato è un assioma.

$$\forall x_1 \ldots \forall x_n \forall y_1 \ldots \forall y_n (x_1 = y_1 \wedge \cdots \wedge x_n = y_n \rightarrow (R(x_1, \ldots, x_n) \leftrightarrow R(y_1, \ldots, y_n)).$$

(12)

$$\forall x \forall y \forall z ((x = x) \wedge ((x = y) \rightarrow (y = x)) \wedge ((x = y) \wedge (y = z)) \rightarrow (x = z))).$$

Definiamo due Regole di Deduzione.

- Modus Ponens: Da F e $(F \rightarrow G)$ si deduce G.
- Generalizzazione: Da F si deduce $\forall x F$.

Osservazione 1.1. La regola di Generalizzazione può sembrare a prima vista bizzarra. Va intesa nel modo seguente: è pratica matematica comune dimostrare un enunciato universale considerando un *generico* elemento del dominio di interesse; per esempio un generico numero naturale n. Questo è esattamente quanto la regola di Generalizzazione esprime: se ho dimostrato una formula F contenente una variabile libera x questa variabile è una *variabile generica* di cui non assumo alcuna proprietà specifica; è dunque corretto concludere la generalizzazione universale. Altra cosa è dire che una formula $F(x) \rightarrow \forall x F(x)$ è dimostrabile: ovviamente una formula di questo tipo è in generale lungi dall'essere logicamente valida!

Introduciamo il concetto di dimostrazione in modo del tutto analogo a quanto fatto nel caso della logica proposizionale.

Definizione 1.2 (Dimostrazione/Deduzione)**.** Una dimostrazione nel calcolo dei predicati è una sequenza finita di formule $(F_1, \ldots, F_n)$, dove per ogni i vale

- F_i è un assioma, oppure
- Esiste $j < i$ tale che $F_i = \forall x F_j$. (Generalizzazione)

- Esistono $j, k < i$ tali che F_k è $F_j \to F_i$. (Modus Ponens)

Una formula F è dimostrabile se esiste una dimostrazione $(F_1, \ldots, F_n)$ nel calcolo dei predicati tale che $F_n = F$. Indichiamo questo fatto con $\vdash F$. Una formula dimostrabile è anche detta un teorema logico.

Il sistema formale (insieme di assiomi e regole di inferenza algoritmicamente decidibili) appena definito è detto il **Calcolo dei Predicati del I ordine (con uguaglianza)**. Per la precisione, ad ogni linguaggio $\mathcal{L}$ fissato possiamo associare *un* calcolo dei predicati del I ordine restringendoci a considerare formule in $\mathcal{L}$.

Il Calcolo dei Predicati intende catturare la nozione di validità logica. I teoremi dimostrabili usando solo gli assiomi del Calcolo dei Predicati sono teoremi di logica pura. Il concetto di dimostrazione si estende facilmente a dimostrazioni matematiche basate su premesse non logiche.

Se rilassiamo la definizione di dimostrazione accettando che F_i possa essere una formula in un certo insieme Γ, otteniamo la nozione di derivazione di una formula F da un insieme di formule Γ. Denotiamo questa relazione con $\Gamma \vdash F$. Diciamo che F è dimostrabile da Γ o anche che F è un teorema di Γ. Anche se Γ è un insieme infinito, una dimostrazione da Γ è sempre un oggetto finito che coinvolge un numero finito di formule in Γ.

Definizione 1.3 (Dimostrazione da Premesse)**.** Sia Γ un insieme di formule. Una dimostrazione da premesse in Γ è una sequenza di formule $(F_1, \ldots, F_n)$, dove per ogni i vale

- F_i è un assioma, oppure
- F_i è un elemento di Γ, oppure
- Esiste $j < i$ tale che $F_i = \forall x F_j$. (Generalizzazione)
- Esistono $j, k < i$ tali che F_k è $F_j \to F_i$. (Modus Ponens)

Una formula F è dimostrabile da Γ se esiste una dimostrazione $(F_1, \ldots, F_n)$ con premesse in Γ e tale che $F_n = F$. Indichiamo questo fatto con $\Gamma \vdash F$. Diciamo anche che F è un teorema di Γ.

2. Proprietà fondamentali del Calcolo dei Predicati

Sia P una formula proposizionale scritta nelle variabili proposizionali $p_1, \ldots, p_n$. Siano $F_1, \ldots, F_n$ formule del I ordine. Sia F la formula del I ordine ottenuta sostituendo in P la variabile p_i con la formula F_i. Se P è una tautologia allora F è detta una istanza al I ordine di una tautologia proposizionale.

Osservazione 2.1. Ogni istanza di una tautologia proposizionale è un teorema del Calcolo dei Predicati.

Questa osservazione si basa sul fatto che gli Assiomi Proposizionali sono completi per le tautologie proposizionali. Una formula proposizionale (costruita da variabili proposizionali usando i connettivi booleani) è una tautologia (i.e., è vera per tutti gli assegnamenti di valori booleani alle sue variabili proposizionali) se e solo se è derivabile dagli Assiomi Proposizionali.

Osservazione 2.2. Ogni teorema del Calcolo dei Predicati è logicamente valido, ossia: per ogni formula F, se $\vdash F$ allora $\models F$.

Questa osservazione si basa sul fatto che gli assiomi sono logicamente validi e che le regole di inferenza preservano la validità.

Osservazione 2.3. Il Calcolo dei Predicati è coerente, ossia per nessuna formula F vale $\vdash F$ e $\vdash \neg F$.

Segue ovviamente dalle precedenti. Da notare che questa dimostrazione della coerenza del Calcolo dei Predicati è tutt'altro che costruttiva, ma si basa sulla nozione (infinitaria) di validità in tutti i modelli. Esistono diverse dimostrazioni puramente sintattiche e induttive di coerenza per il Calcolo dei Predicati.

Osservazione 2.4. L'insieme dei teoremi del Calcolo dei Predicati è algoritmicamente enumerabile.

Possiamo enumerare tutti e soli i teoremi con la seguente procedura meccanica. Fissiamo una enumerazione degli Assiomi del Calcolo dei predicati, $(A_1, A_2, \dots)$. Otteniamo una enumerazione dei teoremi come segue. Mettiamo A_1 in lista. Aggiungiamo tutte le formule ottenute applicando il Modus Ponens o una sola applicazione di Generalizzazione con v_1 come variabile quantificata. Aggiungiamo A_2 alla lista. Aggiungiamo tutte le formule ottenute applicando a formule della nuova lista il Modus Ponens o una applicazione della Generalizzazione con v_1 o v_2 come variabile quantificata. E così via...

3. Teorema di Deduzione

Nel caso predicativo il Teorema di Deduzione *non vale nella sua forma generale.* Consideriamo la formula predicativa $R(v_1)$. Sia $\mathfrak{A}$ una struttura tale che

- Il dominio A di $\mathfrak{A}$ ha almeno due elementi, siano a, b.
- R viene interpretato in $\mathfrak{A}$ come una proprietà $R^{\mathfrak{A}}$ che è soddisfatta soltanto dall'elemento a.

Allora, per ogni assegnamento α tale che $\alpha(v_1) = a$, abbiamo che $\mathfrak{A} \models R(v_1)[\alpha]$. D'altra parte, per ogni assegnamento α, $\mathfrak{A} \not\models ((\forall v_1)R(v_1))[\alpha]$, perché questo vorrebbe dire che per ogni $c \in A$, vale $\mathfrak{A} \models R(v_1)[\alpha\binom{v_1}{c}]$, ossia che ogni $c \in A$ soddisfa il predicato $R^{\mathfrak{A}}$. Ma questo non è vero per scelta di $\mathfrak{A}$. Dunque, $\mathfrak{A}$ è una struttura che non soddisfa l'implicazione $R(v_1) \to (\forall v_1)R(v_1)$. Dunque, in generale, la formula $F \to (\forall v_i)F$ non è una verità logica e dunque non è un teorema del Calcolo dei Predicati. D'altra parte vale sempre $F \vdash (\forall v_i)F$, per la regola di Generalizzazione. Questo dimostra che il Teorema di Deduzione non vale nella sua forma generale. Vale però in una forma più debole.

Teorema 3.1 (Teorema di Deduzione Predicativo)**.** *Sia E un enunciato, G una formula e Γ un insieme di formule. Se*

$$\Gamma, E \vdash G$$

allora

$$\Gamma \vdash E \to G.$$

Dimostrazione. Sia $(D_1, \dots, D_n)$ una dimostrazione di G da Γ, E. Allora $D_n = G$. Per induzione dimostriamo che

$$\Gamma \vdash E \to D_i.$$

L'unico caso che non abbiamo già trattato nella versione proposizionale è quello di una Generalizzazione.

Sia $j < i$ tale che D_i è $\forall x D_j$. Per ipotesi induttiva abbiamo $\Gamma \vdash E \to D_j$. Dato che E è un enunciato, x non è una variabile libera di E. Usiamo l'assioma $(\forall x)(E \to D_j) \to (E \to \forall x D_j)$ e il fatto che, per Generalizzazione da $\Gamma \vdash E \to D_j$ abbiamo $\Gamma \vdash \forall x(E \to D_j)$. □

4. Alcune regole derivate notevoli

Regola dell'istanza Sia t libero per x in $F(x)$. Allora

$$(\forall x)F(x) \vdash F(t).$$

La regola segue direttamente per Modus Ponens da un assioma.

Un caso particolare è quando t è proprio x, e si ha $(\forall x)F(x) \vdash F$.

Regola dell'esistenziale Sia t libero per x in F. Sia $F[x/t]$ ottenuta da F sostituendo tutte le occorrenza libere di x con t (t può anche non apparire in $F[x/t]$). Allora

$$F[x/t] \vdash (\exists x)F.$$

Si dimostra $\vdash F[x/t] \to (\exists x)F$. Si usa l'assioma $(\forall x)\neg F \to \neg F[x/t]$, la tautologia $(A \to \neg B) \to (B \to \neg A)$ e il Modus Ponens, per ottenere $\vdash F[x/t] \to \neg(\forall x)\neg F$.

Un caso particolare è $F(t) \vdash (\exists x)F(x)$, con t libero per x in $F(x)$. Se t è proprio x, abbiamo $F(x) \vdash (\exists x)F(x)$.

LOGICA MATEMATICA

A.A. 23/24, LOGICA PREDICATIVA, DISPENSA N. 7

SOMMARIO. Teorema di Completezza predicativo (per teorie con linguaggio numerabile). Lemma di Lindenbaum, modello di Henkin, teoria con testimoni.

1. Teorema di Completezza

Dimostriamo il Teorema di Completezza di Gödel. Dimostriamo l'equivalenza dei punti (A) e (B) seguenti.

(A) L'enunciato E è deducibile dalla teoria T (i.e., $T \vdash E$).
(B) L'enunciato E è valido in tutti i modelli di T (i.e., $T \models E$).

In particolare, se T è la teoria vuota, abbiamo l'equivalenza tra

$$E \text{ è un teorema del calcolo dei predicati} \Leftrightarrow E \text{ è vero in tutte le strutture.}$$

Osserviamo che

- $\vdash E$ abbrevia una quantificazione esistenziale su un insieme numerabile di oggetti finiti: *Esiste una derivazione formale* $(D_1, \dots, D_n)$ *con conclusione* E.
- $\models E$ abbrevia una quantificazione universale su un insieme non numerabile di oggetti anche infiniti: *Per ogni struttura* $\mathfrak{A}$ *(adeguata per il linguaggio di* E*),* E *è vero in* $\mathfrak{A}$.

Abbiamo inoltre la seguente riformulazione. Sono equivalenti i punti (C) e (D) seguenti:

(C) La teoria T non deduce contraddizioni (i.e., T è coerente).
(D) La teoria T ha un modello (i.e., T è soddisfacibile).

Le due doppie implicazioni ("A se e solo se B" e "C se e solo se D") sono due formulazioni del Teorema di Completezza. Le formulazioni sono equivalenti. In particolare, $A \Rightarrow B$ è equivalente a $D \Rightarrow C$ e $B \Rightarrow A$ è equivalente a $C \Rightarrow D$.

Premettiamo due osservazioni elementari.

Osservazione 1.1. Per definizione, una teoria T è coerente se non esiste un enunciato F tale che $T \vdash F$ e $T \vdash \neg F$. Questo equivale a dire che esiste un enunciato E tale che $T \nvdash E$. In altre parole, una teoria è coerente se e solo se non dimostra tutti gli enunciati (se e solo se esiste un enunciato che la teoria non dimostra). Basta osservare che $(A \to (\neg A \to E))$ è una tautologia (detta *ex falso quodlibet*).

Osservazione 1.2. $T \cup \{E\}$ è coerente se e solo se $T \nvdash \neg E$. Supponiamo $T \cup \{E\}$ coerente e supponiamo per assurdo che $T \vdash \neg E$. Allora $T \cup \{E\} \vdash E \wedge \neg E$ ed è incoerente. Contraddizione. Supponiamo ora che $T \nvdash \neg E$ ma che, per assurdo, $T \cup \{E\}$ è incoerente. Allora $T \cup \{E\} \vdash \neg E$. Dunque $T \vdash E \to \neg E$. Dunque $T \vdash \neg E$ (per logica proposizionale). Questo contraddice l'ipotesi. Contraddizione.

Esercizio: Dimostrare l'equivalenza tra le due formulazioni del Teorema di Completezza.

Appunti preparati da Lorenzo Carlucci, lorenzo.carlucci@uniroma1.it.

L'implicazione difficile è $C \to D$. Si tratta di dedurre da "Non esiste una deduzione formale di una contraddizione da premesse in T", l'esistenza di un insieme con certe proprietà (una struttura che soddisfa T).

2. Estensione Completa

Mostriamo ora che nello spazio delle estensioni di una teoria esiste sempre un oggetto massimale. Da ora in poi restringiamo l'attenzione a teoria in un linguaggio numerabile. Il Teorema di Completezza vale anche per teorie con linguaggi non numerabili ma lo dimostriamo qui solo per teorie con linguaggi numerabili.

Diciamo che una teoria T' *estende* una teoria T se $T \subseteq T'$. Diciamo che una teoria T è *completa* se, per ogni enunciato E nel linguaggio di T, vale $T \vdash T$ oppure $T \vdash \neg E$.

Lemma 2.1 (Lemma di Lindenbaum)**.** *Ogni teoria coerente (in un linguaggio numerabile) ammette un'estensione coerente e completa.*

Dimostrazione. Sia T una teoria coerente nel linguaggio $\mathcal{L}$ (numerabile). Sia $\{E_1, E_2, \dots, \}$ una enumerazione di tutti gli enunciati di $\mathcal{L}$. Sia $S_0 = T$. Dato S_n, per $n \geq 0$, definiamo S_{n+1} come segue.

$$S_{n+1} = \begin{cases} S_n \cup \{E_{n+1}\} & \text{se } S_n \cup \{E_{n+1}\} \text{ è coerente,} \\ S_n & \text{altrimenti.} \end{cases}$$

La condizione $S_n \cup \{E_{n+1}\}$ è coerente è equivalente a $S_n \nvdash \neg E_{n+1}$.

Sia $S_\infty = \bigcup_{n \in \mathbf{N}} S_n$. S_∞ è un insieme di enunciati coerente e completo. (Esercizio).

□

Vediamo che in un senso molto preciso una teoria coerente è completa è *quasi* un modello. Per la discussione che segue ci concentriamo sul Calcolo dei Predicati senza uguaglianza. Facciamo la seguente restrizione. Supponiamo che T sia una teoria coerente e completa *in un linguaggio senza funzioni e con una quantità numerabile di costanti*.

Sia T una teoria coerente e completa. Definiamo una struttura $\mathfrak{M}$ (detto il modello dei termini di T) come segue.

(1) Il dominio M del modello è l'insieme dei termini chiusi nel linguaggio di T.
(2) L'interpretazione di una costante c è data dalla costante c stessa.
(3) L'interpretazione di un simbolo di relazione R è data dall'insieme delle sequenze di termini chiusi di cui T dimostra che soddisfano la relazione, i.e., $(t_1, \dots, t_n) \in R^{\mathfrak{M}}$ se e solo se $T \vdash R(t_1, \dots, t_n)$.
(4) L'interpretazione di un simbolo di funzione f è l'associazione $t_1, \dots, t_d \mapsto f(t_1, \dots, t_d)$.

Osservazione 2.2. Si osserva facilmente che l'intepretazione in $\mathfrak{M}$ di un termine chiuso t coincide con il termine stesso, i.e., $t^{\mathfrak{M}} = t$. (Esercizio).

Per induzione sul numero dei connettivi e dei quantificatori proviamo a dimostrare che, per ogni enunciato E nel linguaggio di T, vale

$$\mathfrak{M} \models E \iff T \vdash E.$$

Nella dimostrazione usiamo in modo essenziale la coerenza e la completezza di T. Vedremo che è possibile trattare tutti i casi tranne quello dei quantificatori.

(Caso 1) Se E è un enunciato atomico $R(t_1, \dots, t_k)$ abbiamo: se $T \vdash R(t_1, \dots, t_k)$ allora $t_1, \dots, t_k$ è in $R^{\mathfrak{M}}$.

(Caso 2) Se E è $\neg G$. Se $\mathfrak{M} \models E$ allora $\mathfrak{M} \nvDash G$ e per ipotesi induttiva $T \nvdash G$. Dato che T è completa segue $T \vdash \neg G$. Se $\mathfrak{M} \nvDash E$, allora $\mathfrak{M} \models G$ e per ipotesi induttiva $T \vdash G$. Dato che T è coerente segue $T \nvdash \neg G$.

(Caso 3) E è $(G \to H)$. Se $\mathfrak{M} \nvDash (G \to H)$ allora $\mathfrak{A} \models G$ e $\mathfrak{M} \nvDash H$. Per ipotesi induttiva $T \vdash G$ e $T \nvdash H$. Per completezza di T segue $T \vdash \neg H$. Usando la tautologia $(G \to (\neg H \to \neg(G \to H)))$ ottengo $T \vdash \neg(G \to H)$. Dato che T è coerente, segue $T \nvdash (G \to H)$. Supponiamo ora che $\mathfrak{M} \models (G \to H)$. Allora: se $\mathfrak{M} \models G$ allora $\mathfrak{M} \models H$. Per ipotesi induttiva, $\mathfrak{M} \models G$ se e solo se $T \vdash G$, e $\mathfrak{M} \models H$ se e solo se $T \vdash H$. Abbiamo quindi che: se $T \vdash G$ allora $T \vdash H$. In genere questo non basta a concludere che $T \vdash G \to H$.

Supponiamo però che $T \nvdash (G \to H)$. Dato che T è completa segue che $T \vdash \neg(G \to H)$. Allora (per logica proposizionale) $T \vdash G$ e $T \vdash \neg H$. Contraddizione.

(Caso 4) Proviamo a considerare il caso di un enunciato predicativo, e.g., $\forall x F$. Abbiamo due casi. F è un enunciato oppure F ha qualche variabile libera. Nel primo caso, vale $\mathfrak{M} \models F$ se e solo se $T \vdash F$. Inoltre $T \vdash F \leftrightarrow \forall x$, e $\mathfrak{M} \models F$ se e solo se $\mathfrak{M} \models \forall x F$.

Consideriamo ora il secondo caso. Se F ha qualche variabile libera, allora F ha x come unica variabile libera (dato che $\forall x F$ è un enunciato).

Consideriamo l'implicazione

$$\text{Se } T \vdash \forall x F(x) \text{ allora } \mathfrak{M} \models \forall x F(x).$$

Supponiamo che $\mathfrak{M} \not\models \forall x F$ ma $T \vdash \forall x F(x)$. Esiste un assegnamento α in M tale che $\mathfrak{M} \models \neg \forall x F(x)[\alpha]$. Dunque esiste un elemento del dominio t (t è un termine chiuso) tale che $\mathfrak{M} \models \neg F(x)[\alpha\binom{x}{t}]$. Allora $\mathfrak{M} \not\models F(t)$, dato che l'interpretazione di t sotto qualunque assegnamento in M è proprio t. D'altra parte $T \vdash \forall x F(x)$ e quindi $T \vdash F(t)$. Per ipotesi induttiva vale $\mathfrak{M} \models F(t)$. Contraddizione.

Non ci resta ora che da dimostrare

$$\text{Se } \mathfrak{M} \models \forall x F \text{ allora } T \vdash \forall x F.$$

Se $\mathfrak{M} \models \forall x F$ allora, per definizione di soddisfazione, per ogni $m \in M$ vale $\mathfrak{M} \models F(x)[\binom{x}{m}]$. Dato che M è l'insieme dei termini chiusi, si ha che per ogni termine chiuso t, $\mathfrak{M} \models F(x)[\binom{x}{t}]$. Si può dimostrare (Esercizio!) che $\mathfrak{M} \models F(x)[\binom{x}{t}]$ se e solo se $\mathfrak{M} \models F(t)$. Dunque abbiamo che per ogni termine chiuso t vale $\mathfrak{M} \models F(t)$. Per ipotesi induttiva segue che per ogni termine chiuso t vale $T \vdash F(t)$. Ma da questo non si può concludere in generale che $T \vdash \forall x F(x)$.

Proviamo invece a ragionare per assurdo: supponiamo $\mathfrak{M} \models \forall x F$ e $T \nvdash \forall x F$. Per completezza di T vale $T \vdash \neg E$. Dunque $T \vdash \exists x \neg F(x)$. Qui non possiamo andare avanti, perché non siamo in grado di ridurci ad una formula di complessità più semplice alla quale poter applicare l'ipotesi di induzione!

Se fossimo in grado di dedurre dal fatto che $T \vdash \exists x \neg F(x)$, che $T \vdash \neg F(t)$ per qualche termine chiuso t, potremmo procedere con la dimostrazione. Vediamo di seguito che è sempre possibile estendere una teoria a una teoria che permette questo passaggio.

Osservazione 2.3. Il fatto che non sia possibile dedurre $T \vdash \forall x F(x)$ sapendo che per ogni termine chiuso t vale $T \vdash F(t)$ può destare qualche perplessità. Consideriamo però il caso seguente: sia P una proprietà dei naturali, e supponiamo che per ogni numero naturale n siamo in grado di dimostrare $P(n)$. Quando questo accade, nella usuale pratica matematica, significa che abbiamo una dimostrazione di $P(n)$ con un qualche grado di *uniformità* in n. Tipicamente abbiamo una dimostrazione per induzione, che ha un altissimo grado di uniformità, essendo basata su una dimostrazione che la proprietà P si trasferisce da un generico n al suo successore immediato. In questo caso siamo legittimati a concludere che abbiamo una dimostrazione del fatto che per ogni n vale $P(n)$. Il caso che stiamo considerando è diverso: l'ipotesi che per ogni t termine chiuso abbiamo una dimostrazione da T di $F(t)$ non dà alcuna garanzia sull'uniformità di queste dimostrazioni. In teoria potrebbero essere estremamente diverse l'una dall'altra al variare di t. Questo dà un'idea del perché in generale non possiamo essere certi che T abbia abbastanza risorse argomentative per "unificare" queste (infinite) dimostrazioni differenti in un unico argomento per $\forall x F(x)$.

3. Teorie con testimoni

Definizione 3.1 (Teoria con testimoni)**.** T è una teoria con testimoni se per ogni formula $F(x)$ con x unica variabile libera esiste un termine chiuso t tale che

$$T \vdash (\exists x) \neg F(x) \to \neg F(t).$$

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

Prima di dimostrare che è possibile estendere ogni teoria coerente a una teoria coerente con testimoni (aggiungendo solo una quantità numerabile di nuovi simboli al linguaggio), facciamo vedere che, se la teoria T è coerente, con testimoni e completa, possiamo concludere la dimostrazione che il modello $\mathfrak{M}$ costruito sopra è un modello di T.

Restava da concludere il caso di un enunciato $\forall x F(x)$ con $F(x)$ aperta.

Dimostriamo che, se $\mathfrak{M} \models \forall x F$, allora $T \vdash \forall x F$. Per assurdo, supponiamo $\mathfrak{M} \models \forall x F$ e $T \nvdash \forall x F$. Per completezza di T vale $T \vdash \neg E$. Dunque $T \vdash \exists x \neg F(x)$. Dato che T è una teoria con testimoni, esiste un termine chiuso t tale che $T \vdash \exists x \neg F(x) \to \neg F(t)$. Dunque $T \vdash \neg F(t)$. Per ipotesi induttiva $\mathfrak{M} \models \neg F(t)$. Ma da $\mathfrak{M} \models \forall x F(x)$, e vale $\mathfrak{M} \models \forall x F(x) \to F(t)$ (vale per qualunque termine chiuso). Dunque $\mathfrak{M} \models F(t)$, una contraddizione con $\mathfrak{M} \models \neg F(t)$.

La dimostrazione è conclusa.

Teorema 3.2. *Per ogni teoria T coerente esiste una teoria T' tale che*

(1) *T' è un'estensione di T,*
(2) *T' è una teoria con testimoni,*
(3) *Il linguaggio di T' è numerabile ed estende quello di T,*
(4) *T' è coerente.*

Dimostrazione. Estendiamo il linguaggio di T con nuove costanti $\{b_1, b_2, \dots\}$. Sia T_0 uguale a T con l'aggiunta di tutti gli assiomi logici nel nuovo linguaggio. Ovviamente T_0 è coerente.

Sia

$$F_1(x_1), F_2(x_2), \dots$$

una enumerazione di tutte le formule di T_0 con una sola variabile libera. Sia

$$b_{j_1}, b_{j_2}, \dots$$

una lista di nuovi simboli di costante tale che

- b_{j_k} non appare in $F_1(x_1), \dots, F_k(x_k)$
- b_{j_k} è diverso da $b_{j_1}, \dots, b_{j_{k-1}}$

Sia W_k l'enunciato seguente

$$\exists x_k \neg F_k(x_k) \to \neg F_k(b_{j_k}).$$

Sia

$$T_n = T_0 \cup \{W_1, \dots, W_n\},$$

e sia

$$T_\infty = \bigcup_n T_n.$$

Dimostriamo che T_∞ è coerente. Basta dimostrare che tutti i T_n sono coerenti.

T_0 è coerente. Supponiamo T_{n-1} coerente e dimostriamo coerente T_n. Se T_n è incoerente, in particolare

$$T_n \vdash \neg W_n.$$

Dunque

$$W_n, T_{n-1} \vdash \neg W_n,$$

e dunque

$$T_{n-1} \vdash W_n \to \neg W_n.$$

Ma allora $T_{n-1} \vdash \neg W_n$, ossia

$$T_{n-1} \vdash \neg(\exists x_n \neg F_n(x_n) \to \neg F_n(b_{j_n})).$$

Si dimostra allora che

$$T_{n-1} \vdash (\exists x_n) \neg F_n(x_n) \quad \text{e} \quad T_{n-1} \vdash \neg\neg F_n(b_{j_n}).$$

Dunque anche $T_{n-1} \vdash F_n(b_{j_n})$. Sia y una variabile che non occorre nella dimostrazione di $F_n(b_{j_n})$ in T_{n-1}. T_{n-1} è $T_0 \cup \{W_1, \dots, W_{n-1}\}$. Le seguenti osservazioni sono fondamentali. La costante b_{j_n} non appare in $W_1, \dots, W_{n-1}$. T_0 contiene T e le istanze di assiomi logici nel linugaggio esteso con le nuovi costanti. La costante b_{j_n} non compare in T. Può comparire in istanze di assiomi logici. In questo caso possiamo osservare che se sostituiamo b_{j_n} con una nova variabile, la formula risultante resterà una istanza di assioma logico. Globalmente, la teoria T_{n-1} non contiene alcuna ipotesi *specifica* sulla costante b_{j_n}. Per questo motivo possiamo concludere che possiamo concludere che

$$T_{n-1} \vdash F_n(y)$$

dove y è una costante che non compare nella dimostrazione e sostituiamo nella dimostrazione ogni occorrenza della costante b_{j_n} con la variabile y (Esercizio: verificare questa osservazione, anche cercando in letteratura). Allora abbiamo

$$T_{n-1} \vdash \forall y F_n(y)$$

e dunque – rinominando le variabili vincolate – abbiamo

$$T_{n-1} \vdash \forall x_n F_n(x_n),$$

dato che x_n è libera per y in $F_n(y)$ e $F_n(y)$ non ha occorrenze libere di x_n. D'altra parte però

$$\begin{aligned} T_{n-1} \vdash \exists x_n \neg F_n(x_n) &\Longrightarrow T_{n-1} \vdash \neg \forall x_n \neg \neg F_n(x_n) \\ &\Longrightarrow T_{n-1} \vdash \neg \forall x_n F_n(x_n). \end{aligned}$$

Dunque T_{n-1} è incoerente. Contraddizione. □

Abbiamo già visto come dimostrare il seguente teorema. Il modello $\mathfrak{M}$ costruito sopra è ovviamente numerabile.

Teorema 3.3. *Sia T una teoria con testimoni coerente e completa in un linguaggio numerabile. Allora T ha un modello numerabile.*

Teorema 3.4 (Esistenza del Modello)**.** *Ogni teoria coerente (in un linguaggio numerabile) ha un modello numerabile.*

Dimostrazione. Da T coerente passiamo a una sua estensione T' coerente e con testimoni. I termini chiusi di T' sono in quantità numerabile. Da T' passiamo a T'' una sua estensione coerente e completa. Dato che il linguaggio non cambia, T'' è una teoria con testimoni. Il modello dei termini di T'' è un modello numerabile di T. □

Nota Bene: quanto sopra dimostrato funziona per la logica del primo ordine senza identità, ossia dove il simbolo $=$ non è un simbolo logico con una specifica regola di soddisfazione. Nella dimostrazione che il modello dei termini soddisfa la teoria T abbiamo infatti verificato solo il caso degli enunciati atomici di tipo relazione, ossia $R(t_1, \ldots, t_n)$ e non di tipo $(t_1 = t_2)$. Se vogliamo ottenere il risultato per la logica con identità, è sufficiente prendere il quoziente del modello dei termini ottenuto sopra modulo la seguente relazione sul suo dominio:

$$t \sim s \text{ se e solo se } T \vdash (t = s).$$

Si dimostra facilmente che, usando il Calcolo dei Predicati con gli Assiomi per l'identità, la relazione $\sim$ è una relazione di equivalenza. Il quoziente del modello dei termini modulo $\sim$ è allora un modello di T anche nel senso della logica con identità.

LOGICA MATEMATICA

A.A. 23/24, LOGICA PREDICATIVA DISPENSA N. 8

SOMMARIO. Il Teorema di Compattezza e alcune sue applicazioni: assiomatizzabilità e non-assiomatizzabilità di proprietà di strutture, e modelli non-standard dell'aritmetica.

1. Teorema di Compattezza

Il Teorema di Completezza ha la seguente conseguenza notevole. Ricordiamo che una teoria T è detta coerente se per nessun enunciato A vale $T \vdash A \wedge \neg A$.

Se tutti i sottinsiemi finiti di un insieme di enunciati sono coerenti, allora tutto l'insieme è coerente.

Questa implicazione è il verso non-banale di una equivalenza che va sotto il nome di Teorema di Compattezza.

Teorema 1.1 (Teorema di Compattezza, versione 1)**.** *Un insieme di enunciati è coerente se e soltanto se ogni suo sottinsieme finito è coerente.*

Dimostrazione. La dimostrazione è ovvia. Sia T coerente e supponiamo che esista un sottinsieme T_0 finito di T non coerente. Allora $T_0 \vdash A \wedge \neg A$ per qualche A, e a fortiori $T \vdash A \wedge \neg A$. Supponiamo ora che ogni sottinsieme finito di T sia coerente. Per assurdo, sia T incoerente. Allora $T \vdash A \wedge \neg A$ per qualche A. Allora – per definizione di dimostrazione formale! – esistono enunciati $E_1, \dots, E_n \in T$ tali che $E_1, \dots, E_n \vdash A \wedge \neg A$. Ma allora $\{E_1, \dots, E_n\}$ è un sottinsieme incoerente. □

In vista del Teorema di Completezza, il Teorema di Compattezza si riformula come segue.

Teorema 1.2 (Teorema di Compattezza, versione 2)**.** *Un insieme di enunciati ha un modello se e soltanto se ogni suo sottinsieme finito ha un modello.*

Dimostrazione. Ovvio dalla versione precedente, dato che T è coerente se e soltanto se T ha un modello, per il Teorema di Completezza. □

Si osserva che il Teorema di Compattezza si può riformulare come segue:

Teorema 1.3 (Teorema di Compattezza, versione 3)**.** *$T \models E$ se e solo se esiste un sottinsieme finito $T_0 \subseteq T$ tale che $T_0 \models E$.*

Dimostrazione. $T \models E$ se e solo se $T \vdash E$ se e solo se $T \cup \{\neg E\}$ non è coerente. □

2. Applicazione I: (non) assiomatizzabilità

Data una proprietà P di strutture, è naturale chiedersi se può essere espressa/definita da enunciati del primo ordine, ossia se esiste un insieme di enunciati T che soddisfa, per ogni struttura $\mathfrak{A}$ la seguente equivalenza:

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

Se un tale insieme T esiste, diciamo che T *assiomatizza* (o *definisce*) la proprietà P e che P è una *proprietà assiomatizzabile* (o *definibile*) al I ordine.

Risulta naturale in molti casi specificare su quale classe di strutture consideriamo la quantificazione universale qui sopra e in quale linguaggio predicativo chiediamo che sia scritto la teoria T. La domanda è allora: sia $\mathcal{C}$ una classe di strutture e sia P una proprietà delle strutture in $\mathcal{C}$ (per esempio: $\mathcal{C}$ è la classe dei

Note preparate da Lorenzo Carlucci, lorenzo.carlucci@uniroma1.it.

gruppi e P è la proprietà di essere abeliano). Esiste una teoria T in un determinato linguaggio predicativo $\mathcal{L}$ tale che per ogni struttura $\mathfrak{A} \in \mathcal{C}$ si abbia

$$\mathfrak{A} \models T \text{ se e solo se } \mathfrak{A} \text{ ha la proprietà } P?$$

In caso affermativo si dice che T assiomatizza P relativamente alla classe $\mathcal{C}$. Ovviamente la classe $\mathcal{C}$ deve contenere strutture in grado di interpretare il linguaggio $\mathcal{L}$ in cui è scritta T. Nel caso più generale $\mathcal{C}$ è la classe di tutte le strutture.

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 tale che, per ogni struttura $\mathfrak{A}$ nella classe $\mathcal{C}$,

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

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

Il Teorema di Compattezza è un valido strumento per dimostrare che una certa proprietà P non è finitamente assiomatizzabile.

2.1. **Due proprietà fondamentali.** Formuliamo due proprietà elementari riguardanti l'assiomatizzabilità che saranno utili nel seguito.

La prima proprietà è analoga a una proprietà degli insiemi calcolabili: se un insieme e il suo complemento sono entrambi algoritmicamente enumerabili, allora l'insieme è decidibile.

Lemma 2.1. *Se una proprietà P e il suo complemento $\neg P$ sono entrambe assiomatizzabili, allora P è finitamente assiomatizzabile.*

Dimostrazione. Esercizio □

Lemma 2.2. *Se la classe dei modelli di una teoria T è finitamente assiomatizzabile,*[1] *allora è assiomatizzata da un sottinsieme finito di T.*

Dimostrazione. Esercizio! □

Osservazione 2.3. Ogni classe $\mathcal{K}$ di strutture determina ovviamente una proprietà di strutture, la proprietà $P_{\mathcal{K}}$ di appartenere a $\mathcal{K}$. Esistono condizioni (algebriche) generali affinché $P_{\mathcal{K}}$ sia finitamente assiomatizzabile (Teorema di Keisler). In questo corso non le studiamo!

2.2. **Esempio 1: Finitezza.** Cominciamo con un esempio molto semplice: la proprietà di *avere dominio finito* non è assiomatizzabile.

La prima osservazione è che, per ogni $n \in \mathbf{N}$, è possibile esprimere al I ordine (in una teoria con l'identità) che esistono almeno n oggetti distinti.

$$A_n := \exists x_1 \dots \exists x_n (\bigwedge_{1 \leq i < j \leq n} x_i \neq x_j).$$

La seconda osservazione è che la teoria T definita come

$$T = \bigcup_{n \in \mathbf{N}} A_n$$

è soddisfatta da tutte e sole le strutture infinite. Nella terminologia introdotta qui sopra questo significa che la proprietà di avere dominio infinito è assiomatizzabile al I ordine nel linguaggio vuoto (ossia contenente solo i simboli logici e il simbolo di identità) relativamente alla classe di tutte le strutture.

Supponiamo che anche la proprietà di essere avere dominio finito sia assiomatizzabile. Sia T' un insieme di enunciati che la assiomatizza. Allora ogni sottinsieme della teoria

$$T \cup T'$$

[1] La classe dei modelli di una teoria T è l'insieme di strutture $\{\mathfrak{A} : \mathfrak{A} \models T\}$. Il Lemma si può riformulare anche così: se T è un insieme infinito di enunciati che assiomatizza la proprietà P, e P è finitamente assiomatizzabile, allora è assiomatizzata da un sottinsieme finito di T. In questo caso la classe dei modelli di T è esattamente la classe delle strutture che soddisfano la proprietà P.

è soddisfacibile, perché esistono insiemi finiti arbitrariamente grandi. Per Compattezza, è assiomatizzabile anche $T \cup T'$. Ma ogni modello di T è infinito. Si osserva che lo stesso risultato si può ottenere aggiungendo un'infinità numerabile di nuove costanti $c_1, c_2, \ldots$ e, per ogni $n \in \mathbf{N}$, la congiunzione $\bigwedge_{i<j\leq n} c_i \neq c_j$.

Abbiamo così dimostrato che la proprietà di avere dominio finito non è assiomatizzabile al I ordine (relativamente alla classe di tutte le strutture e in nessun linguaggio predicativo). (Esercizio: per quali classi di strutture posso affermare la stessa cosa? Esistono classi di strutture in cui la proprietà è assiomatizzabile?).

L'argomento si generalizza come nel seguente teorema.

Teorema 2.4. *Sia T un insieme di enunciati. Se T ha modelli finiti arbitrariamente grandi, allora ha un modello infinito.*

Dimostrazione. L'argomento è identico a quello svolto sopra. □

Abbiamo visto che la proprietà di *essere un insieme infinito* è assiomatizzabile (dalla teoria T di sopra). La proprietà è anche finitamente assiomatizzabile? Ovviamente no, perché altrimenti la negazione degli assiomi sarebbe un'assiomatizzazione finita per la proprietà di essere un insieme finito! (N.B. questo ragionamento vale perché stiamo parlando di una assiomatizzazione finita, che è equivalente a una assiomatizzazione con un unico assioma).

Proposizione 2.5. *La proprietà di essere un insieme infinito è assiomatizzabile ma non è finitamente assiomatizzabile.*

Osservazione 2.6. Un argomento analogo dimostra che la proprietà di *essere un buon ordinamento* non è finitamente assiomatizzabile. Una relazione binaria $<$ su è un buon ordinamento su un insieme X se $<$ è antisimmetrica e tricotomica (i.e., è un ordine totale stretto) e ogni sottinsieme non vuoto di X ha un elemento minimo. Equivalentemente, se non esistono catene infinite discendenti di elementi di X.

2.3. **Esempio 2: Connettività di Grafi.** Un grafo (semplice) $G = (V, E)$ è connesso se e solo se per ogni $v, w \in V$ con $v \neq w$ esiste $n \in \mathbf{N}$ e vertici $x_1, \ldots, x_n \in V$ (tutti distinti tra loro e da v, w) tali che $E(v, x_1), \ldots, E(x_n, w)$ (se $n = 0$ si intende che $E(v, w)$).

Il linguaggio adeguato per parlare di grafi ha un solo simbolo di relazione binario, $E(x, y)$.

Proposizione 2.7. *La connettività non è assiomatizzabile.*

Dimostrazione. Supponi per assurdo che lo sia, e sia T l'assiomatizzazione. La formula seguente dice che non esiste un cammino di lunghezza $\leq n + 1$ tra v e w.

$$P_n(v, w) = \neg(\exists x_1 \ldots \exists x_n)(E(v, x_1) \wedge \bigwedge_{i=1}^{n-1} E(x_i, x_{i+1}) \wedge E(x_n, w)).$$

Consideriamo l'espansione del linguaggio dei grafi con due nuove costanti, a, b. Consideriamo la teoria T' nel nuovo linguaggio definita come segue.

$$T' = T \cup \{P_n(a, b)\}_{n \in \mathbf{N}^+}.$$

Per Compattezza si dimostra che T' è coerente. Un modello di T' non ha cammini tra l'elemento associato ad a e quello associato a b. Dunque è un grafo non connesso, ma soddisfa T. □

Il risultato di sopra può specificarsi meglio come segue: per ogni linguaggio predicativo contenente almeno un simbolo di relazione binaria, non esiste una teoria T che assiomatizza la proprietà di connettività relativamente alla classe di tutte le strutture adeguate per quel linguaggio. Dato che la connettività è una proprietà naturale di grafi il risultato si sintetizza spesso come segue: la connettività non è assiomatizzabile nel linguaggio dei grafi relativamente alla classe dei grafi.

2.4. **Esempio 3: campi.** Il linguaggio adeguato per la teoria dei campi è $\mathcal{L} = \{0, 1, +, \times\}$. I normali assiomi di campo sono al I ordine. Sia F la congiunzione di tali assiomi. Dunque la classe dei campi è assiomatizzabile (finitamente) all'interno delle strutture adeguate per il linguaggio $\mathcal{L}$. Consideriamo ora la classe dei campi di caratteristica[2] p, per p un numero primo.

Proposizione 2.8. *Per ogni $p > 0$, la classe dei campi di caratteristica p è finitamente assiomatizzabile.*

Dimostrazione. Gli assiomi dei campi sono in numero finito. L'assioma

$$C_p := (\underbrace{1 + 1 + \cdots + 1}_{p \text{ volte}} = 0)$$

forza la caratteristica p. □

Proposizione 2.9. *La classe dei campi di caratteristica 0 è assiomatizzabile ma non finitamente assiomatizzabile.*

Dimostrazione. Aggiungendo agli assiomi dei campi l'insieme di assiomi

$$\{\neg C_2, \neg C_3, \ldots, \neg C_p, \ldots\}_{p \in \mathbf{P}}$$

assiomatizzo i campi di caratteristica 0. Se questa teoria fosse assiomatizzabile, allora sarebbe assiomatizzabile da un insieme finito di assiomi che asseriscono che la caratteristica non è $p_1, \ldots, p_k$. Sia p un primo maggiore di tutti i p_i. Allora $\mathbf{Z}_p$ è un modello degli assiomi di campo e degli assiomi $\neg C_{p_1}, \ldots, \neg C_{p_k}$. Ma non è un campo di caratteristica 0. □

Otteniamo come corollario la seguente proposizione, usando una delle due proprietà fondamentali dell'assiomatizzabilità dimostrate sopra.

Proposizione 2.10. *La classe dei campi di caratteristica positiva non è assiomatizzabile.*

Osservazione 2.11. Dai risultati di sopra si può ottenere il seguente Corollario: Sia E un enunciato nel linguaggi dei campi che vale in tutti i campi di caratteristica 0. Allora esiste un numero primo p tale che E vale in tutti i campi di caratteristica $\geq p$.

Osservazione 2.12. Si può dimostrare analogamente che la classe dei campi algebricamente chiusi è assiomatizzabile ma non finitamente assiomatizzabile. Per dimostrare l'assiomatizzabilità basta ricordare che un campo è algebricamente chiuso se e solo se ogni polinimio di grado ≥ 2 a coefficienti nel campo ha uno zero nel campo. Per ogni $k \in \mathbf{N}$ si può esprimere con una formula la relazione $y = x^k$.

Basta partire dalla formula che esprime $xz = y$, denotiamola con $P(z, x, y)$, e porre

$$(\exists z)[z = x^k \wedge P(z, x, y)].$$

La formula appena scritta ha due cariabili libere x, y ed esprime $y = x^{k+1}$. Per $n \geq 2$ sia $T_n(x_0, \ldots, x_n, y, z)$ la formula seguente

$$\begin{aligned}(\exists u_0)\ldots(\exists u_n)(\exists v_0)\ldots(\exists v_n)(\exists w_1)\ldots(\exists w_{n-1}) \\ [1 = u_0 \wedge y = u_1 \wedge y^2 = u_2 \wedge \cdots \wedge y^n = u_n \\ \wedge\, v_0 = u_0 x_0 \wedge v_1 = u_1 x_1 \wedge \cdots \wedge v_n = u_n x_n \\ \wedge\, w_1 = v_0 + v_1 \wedge w_2 = w_1 + v_2 \wedge \cdots \wedge z = w_{n-1} + v_n]\end{aligned}$$

$T_n(x_0, \ldots, x_n, y, z)$ dice che $z = x_0 \cdot y^0 + x_1 y^1 + \cdots + x_n y^n$.

Partendo da questa formula si può formulare, per ogni $n \in \mathbf{N}$ un enunciato τ_n nel linguaggio dei campi che dice che ogni polinomio di grado n ha uno zero nel campo.

Per $n \geq 2$, sia τ_n l'enunciato seguente.

$$(\forall x_0)\ldots(\forall x_n)(\exists y) T_n(x_0, \ldots, x_n, y, 0).$$

[2] La caratteristica di un campo è il minimo n tale che $\underbrace{1 + 1 + \cdots + 1}_{n \text{ volte}} = 0$. Se un tale n esiste è primo, e se non esiste allora si dice che il campo ha caratteristica 0.

τ_n dice che ogni polinomio di grado n ha uno zero nel campo. Allora possiamo assiomatizzare la proprietà di essere un campo algebricamente chiuso con il seguente insieme di enunciati, dove con F indichiamo gli assiomi dei Campi.

$$F \cup \{\tau_n \ : \ 2 \leq n\}.$$

L'aggiunta di tutti i τ_n agli assiomi di campo assiomatizza i campi algebricamente chiusi. Per dimostrare che la classe non è finitamente assiomatizzabile si ragiona come per il caso dei campi di caratteristica positiva, ma qui è più difficile trovare il modello finale. In questo caso si deve usare un risultato di algebra un po' più sofisticato: per ogni $n \in \mathbf{N}$ esiste un campo non algebricamente chiuso in cui tutti i polinomi di grado $\leq n$ hanno uno zero.

3. Applicazione II: Modelli non-standard

3.1. **Campi non-Archimedei.** È noto che i reali soddisfano la seguente proprietà: per ogni $a, b \in \mathbb{R}$ maggiori di 0, esiste un $n \in \mathbf{N}$ tale che $n \times a \geq b$. Questa proprietà è nota come *Assioma di Archimede* e per questo $\mathbb{R}$ è detto un campo archimedeo.

Gli esempi di campi cui siamo più abituati sono tutti archimedei. A priori non è chiaro se possano esistere campi che non soddifano la proprietà di Archimede.

Vediamo come una semplice applicazione della Compattezza è sufficiente a stabilire l'esistenza di questi campi "esotici".

Il linguaggio dei campi è $\{0, 1, \times, +, <\}$. Sia T la teoria di $\mathbb{R}$ in questo linguaggio, ossia l'insieme di tutti gli enunciati soddisfatti in $\mathbb{R}$ con le interpretazioni naturali dei simboli.

Sia c un nuovo simbolo di costante. Per ogni n, denotiamo con $\bar{n}$ il termine chiuso composto da una somma di n simboli 1 (un termine di questo tipo è detto un *numerale*). Consideriamo la seguente estensione di T:

$$T \cup \{\bar{n} < c : \text{ per ogni } n \in \mathbf{N}\}.$$

Si vede facilmente che questa teoria è finitamente soddisfacibile: un suo sottinsieme finito T_0 consiste di un numero finito di assiomi di T e di un numero finito di enunciati atomici di tipo $\bar{n} < c$. Preso m il massimo tale che $\bar{m} < c$ compare in T_0, si verifica facilmente che T_0 è soddisfatto nel modello che ha per dominio i reali, interpreta $+, \times$ e $<$ nel modo naturale e interpreta c in $m + 1$. Per Compattezza l'intera teoria ha un modello. Sia $\mathfrak{A}$ un tale modello. $\mathfrak{A}$ è necessariamente un campo perché soddisfa $T = Th(\mathbb{R})$ (e $\mathbb{R}$ soddisfa tutti gli assiomi di campo). Inoltre $\mathfrak{A} \models 0 < 1$, e $\mathfrak{A} \models 0 < c$. D'altra parte, per ogni $n \ in \mathbf{N}$, $\mathfrak{A} \models \bar{n} \times 1 < c$ e dunque $\mathfrak{A}$ non soddisfa la proprietà di Archimede.

Costruire esplicitamente un campo non-archimedeo non è facile. Esempi importanti di campi di questo tipo sono dovuti a Levi-Civita (campo di Levi-Civita) e a Conway (numeri surreali e iperreali).

3.2. **Modelli non-standard dell'Aritmetica.** Consideriamo il linguaggio $\mathcal{L} = \{0, 1, +, \times, <\}$. La struttura $\mathcal{N} = \{\mathbf{N}, 0, 1, +, \times, <\}$ è una struttura adeguata per $\mathcal{L}$ e $\overline{n}^{\mathcal{N}}$ è n. Questa struttura viene detta *modello standard*.

Sia $Th(\mathcal{N})$ l'insieme degli enunciati in $\mathcal{L}$ veri nella struttura $\mathbb{N}$. Sia c una nuova costante. Consideriamo la teoria

$$T \cup \{\overline{n} < c \ : \ n \in \mathbf{N}\}.$$

Per Compattezza, la teoria T è coerente. In ogni struttura $\mathfrak{A}$ che soddisfa T, l'elemento che interpreta la costante c è maggiore (nel senso di $<^{\mathfrak{A}}$) di tutti gli elementi che interpretano i termini $\overline{n}$. Inoltre, tutte gli enunciati veri nel modello standard $\mathcal{N}$ sono anche veri in $\mathfrak{A}$. Un modello di questo tipo è detto *non-standard*.

Dato un modello non-standard $\mathfrak{A}$, chiamiamo *numeri standard* di $\mathfrak{A}$ le interpretazioni dei numerali, ossia

$$\{a \in A \ : \ \exists n \in \mathbf{N}(a = \overline{n}^{\mathfrak{A}})\}.$$

Vedremo che in generale un modello non-standard di $Th(\mathbf{N})$ inizia con un insieme isomorfo a $\mathbf{N}$, ossia con una copia di $\mathbf{N}$, ma prosegue diversamente!

Dato che $\mathfrak{A} \models Th(\mathbf{N})$, abbiamo che

$$\mathfrak{A} \models \forall x \forall y \forall z (x < y \wedge y < z \to x < z),$$

e

$$\mathfrak{A} \models \forall x \forall y (x < y \vee x = y \vee y < x).$$

Dunque $<^{\mathfrak{A}}$ è un ordine lineare sul dominio A.

Inoltre vale

$$\mathfrak{A} \models \forall x (0 < x \vee 0 = x),$$

e dunque $0^{\mathfrak{A}}$ è il minimo elemento di A rispetto all'ordine $<^{\mathfrak{A}}$.

Dato che vale, per ogni $n \in \mathbf{N}$,

$$\mathfrak{A} \models \neg \exists x (\overline{n} < x \wedge x < \overline{n+1}),$$

abbiamo che non ci sono elementi di A tra due numeri standard.

Dato che vale

$$\mathfrak{A} \models \forall x (x \neq 0 \rightarrow \exists y (y + 1 = x)),$$

abbiamo che ogni elemento di A ha un *predecessore immediato*.

Dato che, per ogni $n \in \mathbf{N}$, vale

$$\mathfrak{A} \models \overline{n} + 1 = \overline{n+1},$$

abbiamo che il successore di un numero standard è un numero standard.

Dunque il modello $\mathfrak{A}$ inizia con una copia di $\mathbf{N}$ (la mappa $n \mapsto \overline{n}^{\mathfrak{A}}$ è un isomorfismo d'ordine tra $\mathbf{N}$ e il segmento iniziale di A costituito dai numeri standard di A). Inoltre A contiene almeno un elemento maggiore di tutti i numeri standard, ossia un non-standard (per es. l'interpretazione di c). Inoltre, se $a \in A$ è un numero non-standard, allora anche il predecessore di a è un numero non-standard. Un elemento non-standard ha dunque infiniti successori e anche infiniti predecessori, nessuno dei quali può essere standard. Intorno a $c^{\mathfrak{A}}$ si sviluppa dunque una copia isomorfa a $\mathbb{Z}$. In realtà si può dimostrare molto di più: in qualunque modello non-standard dell'aritmetica esiste una quantità numerabile di copie di $\mathbb{Z}$ di questo genere e queste copie sono ordinate tra loro in un ordine denso!

Sotto ogni numero non-standard $a \in A$ esiste una catena infinita discendente rispetto all'ordine $<^{\mathfrak{A}}$ di numeri non standard:

$$a > a_1 > a_2 > \dots$$

Dunque l'ordine $<^{\mathfrak{A}}$ non è un buon ordinamento! In particolare in esso non vale il Principio del Minimo Numero, ossia non è vero che ogni sottinsieme non vuoto di A possiede un minimo relativamente all'ordine $<^{\mathfrak{A}}$.

Si osserva però che ogni sottinsieme di A definibile da una formula ha un minimo. Un insieme $X \subseteq A$ è detto definibile da una formula se esiste una formula $F(x)$ con unica variabile libera x tale che

$$X = \{a \in A : \mathfrak{A} \models F(a)\}.$$

Dato che si può esprimere facilmente il Principio del Minimo Numero *relativamente alla formula* F, per esempio come segue:

$$\exists x F(x) \rightarrow \exists x (F(x) \wedge \forall y (F(y) \rightarrow x \leq y)),$$

e dato che questo enunciato, per ogni formula F, è valido nel modello standard, esso è valido anche in $\mathfrak{A}$. Dunque X ha un minimo in A rispetto a $<^{\mathfrak{A}}$.

Da questa osservazione segue però che l'insieme dei numeri standard *non è definibile*! In altre parole non esiste una formula $F(x)$ tale che

$$\{a \in A : \text{ esiste } n \in \mathbf{N} \text{ t.c.} \bar{n}^{\mathfrak{A}} = a\} = \{a \in A : \mathfrak{A} \models F(x) \left[\binom{x}{a}\right]\}.$$

Se $F(x)$ definisse in questo senso i numeri standard in $\mathfrak{A}$ allora $\neg F(x)$ definisce i non-standard:

$$\{a \in A : \text{ per ogni } n \in \mathbf{N} \text{ t.c.} \bar{n}^{\mathfrak{A}} \neq a\} = \{a \in A : \mathfrak{A} \models \neg F(x) \left[\binom{x}{a}\right]\}.$$

Ma il Principio del Minimo Numero per $\neg F$ è valido in $\mathfrak{A}$, ossia

$$\mathfrak{A} \models \exists x \neg F(x) \rightarrow \exists x (\neg F(x) \wedge \forall y (\neg F(y) \rightarrow x \leq y)).$$

Allora dovrebbe esistere un minimo numero non-standard. Ma abbiamo visto sopra che non esiste.

Le osservazioni di sopra escludono che si possa definire una teoria T che catturi (assiomatizzi) $Th(\mathcal{N})$ in modo tale che valga, come per DLO che tutti i modelli numerabili della teoria T sono isomorfi. Resta dunque preclusa la via seguita per dimostrare che DLO (introdotta come assiomatizzazione di $Th(\mathcal{Q})$) era una teoria completa e decidibile. Resta aperta però la possibilità di trovare un insieme di assiomi T con insieme dei teoremi computabilmente enumerabile 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.

4. Applicazione III: dimostrazioni per Compattezza (Extra)

Il Teorema di Compattezza di può usare come strumento per dimostrare risultati di matematica classica in diverse aree, come già visto nel caso della Logica Proposizionale.

Il Teorema di Ramsey è un risultato fondamentale in Combinatoria, con molte applicazioni in diverse aree della Matematica e dell'Informatica.

Teorema 4.1 (Teorema di Ramsey Infinito)**.** *Per ogni colorazione in due colori delle coppie non-ordinate di numeri naturali $f : [\mathbf{N}]^2 \to 2$ esiste un sottinsieme infinito H di $\mathbf{N}$ tale che f ristretta alle coppie di elementi di H (i.e., a $[H]^2$) è costante.*

Un insieme H come sopra è detto *monocromatico* o *omogeneo*. In termini di grafi il teorema afferma che ogni colorazione in due colori degli archi del grafo completo sui naturali ammette una cricca infinita monocromatica. Alternativamente il teorema può riformularsi come: ogni grafo numerabile contiene una cricca infinita o un insieme infinito di vertici indipendenti.

Dimostrazione. Sia $x_0 = 0$. Partizioniamo $\mathbf{N} \setminus \{0\}$ in due parti

$$X = \{x > x_0 \,:\, f(x_0, x) = 0\}$$

$$Y = \{x > x_0 \,:\, f(x_0, x) = 1\}$$

Poiché $X \cup Y$ è infinito, almeno uno tra X e Y è infinito. Scegliamo un tale insieme e chiamiamolo A_1. Sia x_1 il minimo di A_1. Partizioniamo $A_1 \setminus \{x_1\}$ in due parti come segue:

$$X = \{x \in A_1 \,:\, x > x_1 \wedge f(x_1, x) = 0\}$$

$$Y = \{x \in A_1 \,:\, x > x_1 \wedge f(x_1, x) = 1\}$$

Come sopra, almeno una delle due parti è infinita. Selezioniamo una tale parte e chiamiamola A_2. Sia x_2 il suo minimo. Etc.

Procedendo così otteniamo una sequenza infinita di insiemi

$$\mathbf{N} \supseteq A_1 \supseteq A_2 \supseteq A_3 \supseteq \ldots$$

e una sequenza infinita crescente dei corrispondenti minimini

$$\sigma = x_0 < x_1 < x_2 < x_3 < \ldots.$$

La sequenza σ soddisfa la seguente proprietà: la colorazione f di coppie di suoi elementi dipende soltanto dal minimo della coppia: se $i < j < h$ allora $f(x_i, x_j) = f(x_i, x_h)$. In altre parole la sequenza induce una colorazione g di $\mathbf{N}$ in 2 colori, ponendo $g(i) = c_i$ dove $c_i \in \{0, 1\}$ è l'unico colore assegnato da f a tutte le coppie di forma $\{x_i, x_j\}$ con $i < j$. Per il Principio dei Cassetti Infinito infiniti x_i in σ hanno lo stesso colore. Si verifica allora facilmente che la colorazione f è costante su tutte le coppie di tali elementi. □

Si osserva facilmente che la dimostrazione qui sopra funziona non solo per $\mathbf{N}$ ma per ogni insieme numerabile X con la stessa dimostrazione (si parte prendendo $x_0 = \min(X)$).

Il Teorema di Ramsey ha anche una versione finita. La formuliamo qui sotto e la dimostriamo usando la versione infinita e il Teorema di Compattezza.

Teorema 4.2 (Teorema di Ramsey Finito)**.** *Per ogni $m \in \mathbf{N}$ esiste un n che soddisfa quanto segue: Per ogni colorazione delle coppie non-ordinate di naturali in $\{1, \ldots, n\}$ in due colori $f : [\{1, \ldots, n\}]^2 \to 2$ esiste un sottinsieme H of $\{1, \ldots, n\}$ di cardinalità m tale che f ristretta alle coppie di elementi di H (i.e., a $[H]^2$) è costante.*

Dimostrazione. Ragioniamo per assurdo. Supponiamo che il Teorema sia falso. Allora esiste un m tale che nessun n soddisfa la conclusione relativamente a n. Ciò significa che per ogni n esiste una colorazione f_n delle coppie in $\{1, \ldots, n\}$ tale che nessun sottinsieme di $\{1, \ldots, n\}$ di cardinalità m è monocromatico per f_n.

I.e., per ogni sottinsieme S di $\{1, \ldots, n\}$, se S ha cardinalità m allora contiene a, b, c, d tali che $f_n(a, b) \neq f_n(c, d)$. Abbiamo dunque una *sequenza di controesempi* alla validità del teorema, indicizzata da $n \in \mathbf{N}$.

Usiamo il Teorema di Compattezza per incollare tutti questi controesempi finiti in una colorazione di tutte le coppie di naturali che contraddice la versione infinita del Teorema di Ramsey, ossia per cui non esiste un sottinsieme infinito monocromatico.

A questo scopo consideriamo il linguaggio $\mathcal{L}$ composto da un simbolo di relazione binaria R e da infinite costanti $c_1, c_2, \ldots$. Il significato intuitivo di $R(x, y)$ è che la coppia $\{x, y\}$ ha colore 1. Si osservi che in ogni struttura adeguata per $\mathcal{L}$ l'interpretazione del simbolo R è una colorazione in due colori delle coppie non-ordinate di elementi del dominio.

Fissiamo m come sopra. Per ogni n definiamo un enunciato F_n in $\mathcal{L}$ che esprime il fatto che $c_1, \ldots, c_n$ formano un insieme di cardinalità n che non contiene sottinsiemi omogenei di dimensione m (relativamente alla colorazione indotta da R). Dobbiamo esprimere i due fatti seguenti:

(1) $c_1, \ldots, c_n$ sono due a due distinte,
(2) $\{c_1, \ldots, c_n\}$ non contiene un sottinsieme monocromatico di dimensione m.

Entrambi i fatti si esprimono facilmente con enunciati in $\mathcal{L}$ (Esercizio). Per scelta di m abbiamo che la teoria $T = \{F_n : n \in \mathbf{N}\}$ è finitamente soddisfacibile. Infatti, sia T_0 un sottinsieme finito di T. Sia n_0 il massimo tale che F_{n_0} compare in T_0. Consideriamo la colorazione $f_{n_0} : [\{1, \ldots, n_0\}]^2 \to 2$ che dà un controesempio al Teorema Finito di Ramsey per il valore m. Otteniamo una struttura che soddisfa T_0 scegliendo come dominio $\{1, \ldots, n_0\}$, come interpretazione di c_i il numero i per $1 \leq i \leq n_0$, e interpretando R come la colorazione f_{n_0} (i.e. $\{i, j\}$ sono nell'interpretazione di R sse $f_{n_0}(i, j) = 1$.

Per Compattezza la teoria T ha un modello. Sia $\mathfrak{A} \models T$. Consideriamo l'insieme delle interpretazioni delle costanti in $\mathfrak{A}$, i.e.

$$C = \{c_1^{\mathfrak{A}}, c_2^{\mathfrak{A}}, c_3^{\mathfrak{A}}, \ldots, \}.$$

Questo insieme è infinito per le condizioni sulle c_i espresse in T. D'altra parte $R^{\mathfrak{A}}$ è una colorazione di $[A]^2$ in due colori. Applicando il Teorema Infinito di Ramsey a C e $R^{\mathfrak{A}}$ otteniamo un sottinsieme infinito $H \subseteq C$ tale che tutte le coppie scelte in H hanno lo stesso colore nella colorazione indotta da $R^{\mathfrak{A}}$.

Sia k così grande che $A \cap \{c_i^{\mathfrak{A}} : i < k\}$ ha almeno m elementi. Allora la struttura $\mathfrak{A}$ non soddisfa l'enunciato F_k della teoria T. Contraddizione. □

LOGICA MATEMATICA

A.A. 23/24, LOGICA PREDICATIVA DISPENSA N. 9

Sommario. Funzioni e relazioni calcolabili. Rappresentabilità di relazioni e funzioni calcolabile nel modello dei numeri naturali.

1. Funzioni calcolabili

Fissiamo la definizione di funzione (parziale) calcolabile che useremo per dimostrare il teorema di caratterizzazione. Le funzioni calcolabili sono tutte funzioni (possibilmente paziali) di tipo $\mathbf{N}^k \to \mathbf{N}$ per qualche $k \in \mathbf{N}$. Usiamo una definizione particolarmente comoda delle funzioni calcolabili.

Definizione 1.1 (Funzioni Calcolabili)**.** La classe delle funzioni (parziali) calcolabili è la minima classe di funzioni di tipo $\mathbf{N}^k \to \mathbf{N}$ (con $k \in \mathbf{N}$) contenente l'addizione, la moltiplicazione, le proiezioni, la caratteristica del predicato di identità numerica, e chiusa sotto composizione e minimalizzazione.

Le proiezioni sono le funzioni $\pi^n_i : \mathbf{N}^n \to \mathbf{N}$ con $i \leq n$ tali che $\pi^n_i(x_1, \ldots, x_n) = x_i$ (selezione dell'i-esimo elemento di un vettore di n naturali).

La funzione caratteristica del predicato di identità numerica è la funzione $i : \mathbf{N}^2 \to \mathbf{N}$ tale che $i(x, y) = 1$ se $x = y$ e $i(x, y) = 0$ se $x \neq y$.

Per composizione intendiamo la seguente versione generalizzata della normale composizione di funzioni: Se $\theta_1, \ldots, \theta_m$ sono funzioni di tipo $\mathbf{N}^k \to \mathbf{N}$ e $\psi : \mathbf{N}^m \to \mathbf{N}$, la funzione composta è $\varphi : \mathbf{N}^k \to \mathbf{N}$ definita come segue: per ogni $x_1, \ldots, x_k \in \mathbf{N}$:

$$\varphi(x_1, \ldots, x_k) = \psi(\theta_1(x_1, \ldots, x_k), \ldots, \theta_m(x_1, \ldots, x_k)).$$

L'operatore di minimalizzazione è definito come segue. Se $g(\vec{x}, y)$ è una funzione, la funzione $h(\vec{x})$ indicata con $\min z(g(\vec{x}, z) = 0)$ è definita come il minimo z tale che i valori $g(\vec{x}, 0), g(\vec{x}, 1), \ldots, g(\vec{x}, z-1)$ sono definiti e diversi da 0 e $g(\vec{x}, z)$ è definito e uguale a 0. Se un tale z non esiste la funzione è indefinita.

Osservazione 1.2. L'operatore di minimo corrisponde a una ricerca illimitata e possibilmente non terminante. La classe delle funzioni calcolabili contiene pertanto funzioni parziali (usiamo le lettere greche φ, ψ, θ, etc. per indicare funzioni parziali e le lettere latine $f, g, h, \ldots$ per indicare funzioni totali). In generale è possibile dimostrare che nessuna lista di funzioni totali $(f_i)_{i \in \mathbf{N}}$ tale che l'operazione $i \mapsto f_i(i)$ sia effettiva (algoritmica) può esaurire la classe delle funzioni algoritmicamente calcolabili. Infatti la funzione $i \mapsto f_i(i) + 1$ è (informalmente) algoritmicamente calcolabile ma non appartiene alla lista.

Osservazione 1.3. Come discusso a lezione ci sono buoni motivi per identificare la classe $\mathcal{C}$ con la classe delle funzioni calcolabili da un algoritmo nel senso intuitivo. Useremo questa identificazione (nota come Tesi di Church-Turing) per giustificare l'appartenenza di una certa funzione $\varphi : \mathbf{N}^k \to \mathbf{N}$ in base al fatto che la funzione risulta intuitivamente calcolabile da un algoritmo.

2. Teorema di Definibilità

Indichiamo con $\mathcal{N}$ (e a volte con $\mathbf{N}$) la struttura $(\mathbf{N}, 0, 1, +, \times, <)$ adeguata per il linguaggio dell'aritmetica $\mathcal{L} = \{0, 1, +, \times, <\}$; ossia la struttura standard dei numeri naturali.

Note preparate da Lorenzo Carlucci, lorenzo.carlucci@uniroma1.it.

Definizione 2.1 (Definibilità/rappresentabilità/esprimibilità di una funzione nella struttura **N**)**.** Diciamo che φ è *definibile* (o *rappresentabile* p *esprimibile*) in $\mathcal{N}$ da una formula $F(x_1, \ldots, x_k, y)$ (con esattamente $k+1$ variabili libere) se, per ogni $a_1, \ldots, a_k, b \in \mathbf{N}$, vale

$$\varphi(a_1, \ldots, a_k) = b \iff \mathcal{N} \models F\left[\binom{x_1, \ldots, x_k, y}{a_1, \ldots, a_k, b}\right],$$

i.e., se le $k+1$-ple $(a_1, \ldots, a_k, b)$ appartenenti al grafico di φ sono esattamente quelle che soddisfano in $\mathcal{N}$ la formula $F(x_1, \ldots, x_k, y)$ assegnando a_i a x_i e b a y.

Si ricorda che nel linguaggio dell'aritmetica ogni numero naturale $n \in \mathbf{N}$ ha un nome canonico (detto *numerale*), costituito dalla somma di n volte la costante 1. Questo nome proprio del numero n viene denotato con $\bar{n}$.

Dunque nella struttura $\mathcal{N}$, vale che $\mathcal{N} \models F\left[\binom{x_1,\ldots,x_k,y}{a_1,\ldots,a_k,b}\right]$ e solo se $\mathcal{N} \models F(\bar{a_1}, \ldots, \bar{a_k}, \bar{b})$, dove con $F(\bar{a_1}, \ldots, \bar{a_k}, \bar{b})$ l'enunciato ottenuto dalla formula $F(x_1, \ldots, x_k, y)$ sostituendo $\bar{a_i}$ a ogni occorrenza libera di x_i e $\bar{b}$ a ogni occorrenza libera di y.

Ci interessa in particolare la definibilità di funzioni nella struttura $\mathcal{N} = (\mathbf{N}, 0, 1, +, \times, <)$. Da ora in poi useremo **N** per indicare la struttura $\mathcal{N}$ per non appesantire la notazione. Inoltre, se $F(x_1, \ldots, x_n)$ è una formula e $a_1, \ldots, a_n$ sono naturali, scriveremo $F(a_1, \ldots, a_n)$ per indicare l'enunciato $F(\bar{a_1}, \ldots, \bar{a_n})$, dove $\bar{a_i}$ è il termine chiuso costituito dalla somma di a_i costanti 1. Si noti che vale $\mathbf{N} \models F(x_1, \ldots, x_n)[\binom{x_1,\ldots,x_n}{a_1,\ldots,a_n}]$ se e solo se $\mathbf{N} \models F(a_1, \ldots, a_n)$.

Teorema 2.2 (Teorema di Definibilità)**.** *Le funzioni calcolabili sono definibili/rappresentabili in* **N**.

Dimostrazione. Procediamo per induzione, dimostrando che le funzioni di base sono definibili e che l'insieme delle funzioni definibili in $\mathcal{N}$ è chiuso per composizione e minimalizzazione.

Claim 2.3 (Funzioni di Base)**.** *Le funzioni di base sono definibili in* **N**.

Dimostrazione. 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 $\pi^n_i(x_1, \ldots, x_n) = x_i$ (dove $i \in [1, n]$) è definibile dalla formula $H(x_1, \ldots, x_n, z) := (x_1 = x_1 \wedge \cdots \wedge x_i = z \wedge \cdots \wedge x_n = x_n)$, la funzione caratteristica dell'uguaglianza dalla formula $I(x, y, z) := (x = y \wedge z = 1) \vee (x \neq y \wedge z = 0)$. □

Claim 2.4 (Chiusura per Composizione)**.** *Le funzioni definibili in* **N** *sono chiuse per composizione.*

Dimostrazione. Sia $G_i(\vec{x}, y_i)$ una formula che definisce θ_i, $i \in [1, m]$, e sia $H(y_1, \ldots, y_m, z)$ una formula che definisce ψ. Allora

$$\exists y_1 \ldots \exists y_m (G_1(\vec{x}, y_1) \wedge \cdots \wedge G_m(\vec{x}, y_m) \wedge H(y_1, \ldots, y_m, z))$$

definisce $\varphi(\vec{x}) = \psi(\theta_1(\vec{x}), \ldots, \theta_m(\vec{x}))$.

Se $\varphi(\vec{k}) = p$ allora esistono $q_1, \ldots, q_m$ tali che $\theta_i(\vec{k}) = q_i$ e $\psi(q_1, \ldots, q_m) = p$. Dunque $\mathbf{N} \models G_i[\vec{k}, q_i]$ e $\mathbf{N} \models H[q_1, \ldots, q_m, p]$ per ipotesi su G_i e H. □

Claim 2.5 (Chiusura per Minimo)**.** *Le funzioni definibili in* **N** *sono chiuse per minimo.*

Dimostrazione. Sia $\varphi(\vec{x})$ definita come $(\min z)(\psi(\vec{x}, z) = 0)$. Sia $F_\psi(\vec{x}, z, y)$ una formula che definisce il grafico di ψ. Allora la formula seguente definisce il grafico di φ.

$$(\forall w)((w \leq z) \to (\exists y)(F_\psi(\vec{x}, w, y) \wedge (y = 0 \leftrightarrow w = z))).$$

□

□

Estendiamo la nozione di *calcolabile* e di *rappresentabile* da funzioni a relazioni in modo naturale.

Definizione 2.6 (Relazione Calcolabile)**.** Sia $n \geq 1$. Una relazione $R \subseteq \mathbf{N}^n$ è calcolabile se e solo se la sua funzione caratteristica è calcolabile.

Si nosti che il caso $n = 1$ definisce la nozione di insieme (i.e., relazione a un argomento) calcolabile. Questa nozione giustifica domande del tipo: l'insieme dei numeri primi è calcolabile? L'insieme dei quadrati perfetti è calcolabile? etc.

Analogamente estendiamo la nozione di definibilità in $\mathbf{N}$ da funzioni a relazioni come segue.

Definizione 2.7 (Relazione Definibile/rappresentabile/esprimibile in $\mathbf{N}$)**.** Sia $n \geq 1$. Una relazione $R \subseteq \mathbf{N}^n$ è definibile (o rappresentabile, o esprimibile) in $\mathbf{N}$ se e solo se esiste una formula $F(x_1, \ldots, x_n)$ nel linguaggio dell'aritmetica con n variabili libere tali che, per ogni $a_1, \ldots, a_n \in \mathbf{N}$:

(1) Se $(a_1, \ldots, a_n) \in R$ allora $\mathbf{N} \models F(a_1, \ldots, a_n)$, e
(2) Se $(a_1, \ldots, a_n) \notin R$ allora $\mathbf{N} \models \neg F(a_1, \ldots, a_n)$.

Questa nozione giustifica domande del tipo: l'insieme dei numeri primi è definibile? La risposta è sì, considerando la seguente formula $P(x)$ con una variabile libera nel linguaggio dell'aritmetica: $(x \neq 1 \wedge \forall y \forall z(y \times z = x \rightarrow (y = 1 \vee z = 1)))$. La seguente formula $Q(x)$ definisce in $\mathbf{N}$ l'insieme dei quadrati perfetti: $\exists y(x = y \times y)$.

Dalla definizione e dal Teorema di Rappresentabilità per funzioni segue immediatamente il seguente Lemma.

Lemma 2.8. *Le relazioni calcolabili sono rappresentabili/esprimibili in* $\mathbf{N}$.

Dimostrazione. Diamo la dimostrazione per relazioni binarie ($n = 2$) per leggibilità.

Denotiamo con c_R la funzione caratteristica di R, ossia la funzione di tipo $\mathbf{N} \times \mathbf{N} \rightarrow \{0, 1\}$ tale che $c_R(a, b) = 1$ se $(a, b) \in R$ e $c_R(a, b) = 0$ se $(a, b) \notin R$. R è calcolabile sse c_R è calcolabile (per definizione). Se c_R è calcolabile allora è rappresentabile in $\mathbf{N}$. Sia $F(x, y, z)$ la formula che rappresenta c_R. Per definizione di rapprensetabilità abbiamo, per ogni $a, b \in \mathbf{N}$

(1) Se $c_R(a, b) = 1$ allora $\mathbf{N} \models F(a, b, 1)$
(2) Se $c_R(a, b) = 0$ allora $\mathbf{N} \models F(a, b, 0)$.

Dunque abbiamo, dato che $\mathbf{N} \models \neg(0 = 1)$, quanto segue:

(1) Se $c_R(a, b) = 1$ allora $\mathbf{N} \models F(a, b, 1)$
(2) Se $c_R(a, b) = 0$ allora $\mathbf{N} \models \neg F(a, b, 1)$.

Ricapitolando:

(1) Se $(a, b) \in R$ allora $\mathbf{N} \models F(a, b, 1)$
(2) Se $(a, b) \notin R$ allora $\mathbf{N} \models \neg F(a, b, 1)$.

In altre parole, la formula $F(x, y, 1)$ esprime in $\mathbf{N}$ la relazione R. □

I risultati di sopra stabiliscono l'implicazione seguente, per funzioni e per relazioni:

$$\text{Calcolabile} \Rightarrow \text{Esprimibile in } \mathbf{N}.$$

Dunque possiamo dimostrare che una funzione o una relazione non è calcolabile dimostando che non è esprimibile in $\mathbf{N}$.

Osservazione 2.9 (Extra)**.** Si osserva, ispezionando la dimostrazione del Teorema di Rappresentabilità, che le formule usate per rappresentare le funzioni calcolabili hanno tutte una simile struttura sintattica: sono costituite da un prefisso di quantificatori esistenziali seguite da una formula in cui non appaiono quantificatori o appaiono soltanto quantificatori limitati, ossia del tipo seguente: $(\forall x)(x \leq t \rightarrow \ldots)$ oppure $(\exists x)(x \leq t \wedge \ldots)$, dove t è un termine non contenente x. Queste forme vengono dette Σ_1^0. L'unico caso in cui non è evidente è quello della minimalizzazione, dove abbiamo usato la formula:

$$(\forall w)((w \leq z) \rightarrow (\exists y)(F_\psi(\vec{x}, w, y) \wedge (y = 0 \leftrightarrow w = z))).$$

Questa formula non è Σ_1^0 in quanto inizia con un quantificatore universale limitato. Si osserva però facilmente che la formula è equivalente a una Σ_1^0. Sia $F_\psi(\vec{x}, z, y)$ di forma $(\exists v)F'(\vec{x}, z, y, v)$ con $F' \in \Delta_0$. La formula di sopra è equivalente in $\mathbf{N}$ a

$$(\exists u)(\forall w \leq z)(\exists y, v \leq u)(F'(\vec{x}, z, y, v) \wedge (y = 0 \leftrightarrow w = z)).$$

Basta scegliere il testimone di u abbastanza grande da permettere di trovare sotto di esso testimoni per y e v che soddisfano $F'(\vec{x}, z, y, v) \wedge (y = 0 \leftrightarrow w = z)$, per ogni scelta di $w \leq z$. Le due formule risultano

equivalenti nel modello standard. Abbiamo così la seguente versione rinforzata del Teorema di Definibilità: Ogni funzione calcolabile è definibile in $\mathcal{N}$ da una formula Σ_1^0.

Inoltre è possibile dimostrare anche il viceversa, ossia che ogni funzione definibile nei naturali da una formula Σ_1^0 è un una funzione calcolabile. Si ottiene così una completa *caratterizzazione logica* delle funzioni calcolabili.

LOGICA MATEMATICA

A.A. 23/24, LOGICA PREDICATIVA DISPENSA N. 10

Sommario. Indecidibilità algoritmica della verità aritmetica. Risultati di incompletezza.

1. Codifica numerica - Numeri di Gödel

Le funzioni calcolabili e le relazioni calcolabili hanno come argomenti numeri naturali. Risulta però naturale funzioni e relazioni su altri tipi di oggetti tramite codifica con numeri naturali.

In particolare qui ci interessa chiederci se alcuni insiemi di enunciati siano calcolabili o no. Affinché questa domanda abbia senso dobbiamo codificare gli enunciati con numeri naturali.

1.1. **Numeri di Gödel.** In questo contesto, quando ci riferiamo a un insieme S di enunciati ci riferiamo all'insieme dei codici degli enunciati in S. Il primo passo è quello di fissare una codifica numerica dei costrutti linguistici. Questo ci permette di esprimere problemi di decisione e di definire funzioni su simboli, stringhe di simboli, espressioni, formule, sequenze di formule, etc.

Assegnamo a ogni simbolo base del linguaggio (parentesi, connettivi, variabili, simboli di funzione e relazione, costanti) assegnamo un intero positivo dispari.

Il secondo passo è di assegnare un codice a ogni formula ben formata, e.g. $(\exists v_3)((v_3 + 0) < v_1)$. Lo facciamo assegnando un codice a ogni sequenza finita di simboli di base: se $u_0u_1 \dots u_n$ è una tale sequenza, definizamo

$$code(u_0u_1 \dots u_n) = 2^{code(u_0)}3^{code(u_1)} \dots p_n^{code(u_n)},$$

p_i denota l'i-esimo numero primo.

Vogliamo inoltre assegnare un codice numerico a ogni derivazione formale, e per farlo assegnamo un codice a ogni sequenza finita di sequenze finite di simboli di base. Sia $e_0e_1 \dots e_n$ una tale sequenza. Definiamo

$$code(e_0e_1 \dots e_n) = 2^{code(e_0)}3^{code(e_1)} \dots p_n^{code(e_n)}.$$

La funzione *code* è iniettiva e permette di distinguere (algoritmicamente) tra codici di simboli base (dispari), codici di sequenza di simboli base (pari con primo esponente dispari) e codici di sequenze di sequenze di simboli di base (pari con primo esponente pari). Intuitivamente *code* è calcolabile.

Quando qui sotto parliamo di esprimibilità di un insieme di enunciati in una teoria intendiamo l'esprimibilità dell'insieme dei codici numerici associati agli enunciati dell'insieme in questione.

Per esempio, quando consideriamo l'esprimibilità di $Th(\mathbf{N}) = \{E : \mathbf{N} \models E\}$ intendiamo l'esprimibilità dell'insieme $\{code(E) : \mathbf{N} \models E\}$.

1.2. **La funzione diagonale.** Per dimostrare che l'insieme degli enunciati veri nei numeri naturali (la cosidetta *verità aritmetica*) non è decidibile da un algoritmo ci basterà riconoscere che una singola funzione, detta funzione diagonale, che compie una semplice manipolazione sintattiche sulle formule, è calcolabile.

Sia δ la funzione tale che se u è il codice di una formula $A(x)$ con variabile libera x allora $\delta(u)$ è il codice di $A(u)$. δ è ovviamente calcolabile.

Note preparate da Lorenzo Carlucci, lorenzo.carlucci@uniroma1.it.

2. Indecibilità algoritmica dell'aritmetica

Mostriamo ora come ottenere l'indecidibilità algoritmica dell'aritmetica formale usando il Teorema di Rappresentabilità in $\mathbf{N}$ (ogni funzione/relazione calcolabile è rappresentabile/esprimibile in $\mathbf{N}$).

Teorema 2.1 (Tarski)**.** *L'insieme delle verità aritmetiche* $Th(\mathbf{N}) = \{E : \mathbf{N} \models E\}$ *non è esprimibile in* $\mathbf{N}$.

Dimostrazione. Sia $D(x, y)$ la formula che rappresenta la funzione δ in $\mathbf{N}$. Allora vale

$$\delta(k) = j \Longrightarrow \mathbf{N} \models D(k, j).$$

Inoltre, per ogni $k \in \mathbf{N}$,

$$\mathbf{N} \models \forall y \forall z((D(k, y) \wedge D(k, z)) \rightarrow y = z).$$

Supponiamo per assurdo che le verità aritmetiche siano esprimibili in $\mathbf{N}$. Sia $V(y)$ la formula che le esprime. Allora vale, per ogni enunciato H,

$$\mathbf{N} \models H \Longrightarrow \mathbf{N} \models V(code(H))$$

e

$$\mathbf{N} \not\models H \Longrightarrow \mathbf{N} \models \neg V(code(H)),$$

dove indichiamo con $code(E)$ il codice dell'enunciato E.

Consideriamo la formula $A(x)$ seguente.

$$\forall y(D(x, y) \rightarrow \neg V(y)).$$

Sia p il codice di questa formula. Consideriamo $A(p)$, ottenuto sostituendo il numerale p a ogni occorrenza libera di x in $A(x)$.

$$\forall y(D(p, y) \rightarrow \neg V(y)).$$

Sia q il codice di $A(p)$. Per definizione di δ abbiamo $\delta(p) = q$: infatti p è il codice di una formula con una unica variabile libera (la formula $A(x)$, con x come unica variabile libera) e q è il codice dell'enunciato ottenuto sostituendo il numerale p a ogni occorrenza libera di x in $A(x)$.

Dunque, dato che la formula D definisce la funzione δ in $\mathbf{N}$, abbiamo che

$$\mathbf{N} \models D(p, q).$$

Vogliamo ora dimostrare che $\mathbf{N} \models \neg V(q)$. Ragioniamo per casi.

(Caso 1) Se $\mathbf{N} \not\models A(p)$, allora $A(p)$ non è una verità in $\mathbf{N}$ e dunque $code(A(p)) = q \notin Th(\mathbf{N})$. Dunque, per esprimibilità,

$$\mathbf{N} \models \neg V(q).$$

(Caso 2) Se $\mathbf{N} \models A(p)$, allora anche

$$\mathbf{N} \models D(p, q) \rightarrow \neg V(q).$$

Dunque, dato che $\mathbf{N} \models D(p, q)$, segue che

$$\mathbf{N} \models \neg V(q).$$

Abbiamo dimostrato così che $\mathbf{N} \models \neg V(q)$.

Dimostriamo ora che $\mathbf{N} \models V(q)$.

Da $\mathbf{N} \models D(p, q)$ e dalla funzionalità della formula D segue che

$$\mathbf{N} \models D(p, y) \rightarrow y = q.$$

Inoltre vale (dato che $\mathbf{N} \models \neg V(q)$, come sopra dimostrato), abbiamo

$$\mathbf{N} \models y = q \rightarrow \neg V(y).$$

Dunque

$$\mathbf{N} \models D(p, y) \rightarrow \neg V(y).$$

Da ciò segue

$$\mathbf{N} \models \forall y(D(p, y) \rightarrow \neg V(y)),$$

ossia

$$\mathbf{N} \models A(p).$$

Dunque $A(p)$ è una verità aritmetica, ossia $A(p) \in Th(\mathbf{N})$.

Dunque, dato che la formula V rappresenta $Th(\mathbf{N})$, abbiamo che

$$\mathbf{N} \models V(code(A(p))$$

ossia

$$\mathbf{N} \models V(q).$$

Ma questo è impossibile. □

Corollario 2.2 (Indecidiblità della Verità Aritmetica)**.** *L'insieme delle verità aritmetiche* $Th(\mathbf{N}) = \{E : \mathbf{N} \models E\}$ *non è decidibile da un algoritmo.*

Dimostrazione. Abbiamo dimostrato che l'insieme in questione non è esprimibile in $\mathbf{N}$. Ma ogni insieme decidibile è esprimibile in $\mathbf{N}$. □

Corollario 2.3 (Primo Teorema di Incompletezza di Gödel (forma debole))**.** *Se* T *è una teoria decidibile e tale che* $\mathbf{N} \models T$ *allora* T *non è completa: esiste un enunciato* G *tale che* $T \nvdash G$ *e* $T \nvdash \neg G$*.*

Dimostrazione. Se T fosse completa, dato che $\mathbf{N} \models T$ per ipotesi, avremmo che

$$\{E : T \vdash E\} = \{E : \mathbf{N} \models E\},$$

ossia l'insieme dei teoremi di T coinciderebbe con l'insieme delle verità aritmetiche $Th(\mathbf{N})$.

D'altra parte se T è un insieme decidibile di assiomi allora $\{E : T \vdash E\}$ è algoritmicamente decidibile (questo vale in generale). Ma abbiamo dimostrato che $\{E : \mathbf{N} \models E\}$ non è algoritmicamente decidibile. □

LOGICA MATEMATICA

A.A. 23/24, LOGICA PREDICATIVA DISPENSA N. 11

Sommario. Aritmetica Minimale. Rappresentabilità delle funzioni calcolabili in una teoria. Primo Teorema di Incompletezza di Gödel.

Abbiamo dimostrato una forma debole del I Teorema di Incompletezza di Gödel. Vogliamo ora discutere la formulazione originale di questo risultato.

Abbiamo dimostrato la seguente forma debole del I Teorema di Incompletezza di Gödel.

Theorem A (Primo Teorema di Incompletezza di Gödel (forma debole))**.** *Se T è una teoria decidibile (i.e. un insieme decidibile di enunciati) nel linguaggio dell'aritmetica tale che $\mathbf{N} \models T$ allora T è incompleta.*

Vogliamo ora discutere la formulazione originale di questo risultato.

A questo scopo fisseremo una teoria che chiamiamo Aritmetica Minimale e denotiamo con **MA**. **MA** è assiomatizzata da un numero finito di semplici assiomi aritmetici nel linguaggio $\{0, 1, +, \times\}$.

Il Primo Teorema di Incompletezza di Gödel, nella sua formulazione più generale, può allora esprimersi come segue:

Theorem B (Primo Teorema di Incompletezza di Gödel)**.** *Se T è una teoria decidbile (i.e. un insieme decidibile di enunciati) nel linguaggio dell'aritmetica che contiene* **MA** *ed è coerente allora T è incompleta.*

La differenza rispetto al risultato già dimostrato è che non si assume che $\mathbf{N} \models T$, ossia che T sia corretta rispetto alla verità nel modello standard. Questo è rilevante (sopratutto dal punto di vista dei Fondamenti della Matematica) perché le ipotesi su T sono di natura puramente sintattica e non fanno riferimento a strutture infinite e al concetto semantico di soddisfacibilità.

Abbiamo bisogno dei seguenti ingredienti:

(1) Definire una nozione di rappresentabilità (di funzioni e relazioni) reltivamente a una teoria, anziché a una struttura
(2) Dimostrare che la teoria **MA** è sufficiente a rappresentare tutte le funzioni e le relazioni calcolabili.

Con questi ingredienti è possibile dimostrare una versione del Teorema di Tarski più generale. Un altro ingrediente importante dell'approccio originale di Gödel è che fornisce un esempio esplicito di enunciato indimostrabile e irrefutabile relativamente a una data teoria T (decidibile) che estende **MA**.

1. Aritmetica Minimale

A questo scopo fissiamo una teoria che chiamiamo Aritmetica Minimale e denotiamo con **MA**. **MA** è assiomatizzata dai seguenti assiomi nel linguaggio $\{0, 1, +, \times\}$. Denotiamo con **MA** anche la congiunzione della chiusura universale degli assiomi di **MA**. Per ogni $n \in \mathbf{N}$ denotiamo ambiguamente con n il termine ottenuto sommando la costante 1 a se stessa n volte. Con $x \neq y$ abbreviamo $\neg(x = y)$, con $x \leq y$ abbreviamo $(\exists z)(x + z = y)$, e con $x < y$ abbreviamo $x \leq y \wedge x \neq y$.

(Ax 1) $0 + 1 = 1$
(Ax 2) $\forall x(x + 1 \neq 0)$
(Ax 3) $\forall x(x \neq 0 \to \exists z(z + 1 = x))$

Note preparate da Lorenzo Carlucci, lorenzo.carlucci@uniroma1.it.

(Ax 4) $\forall x \forall y(x+1 = y+1 \to x = y)$
(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) $\forall x \forall y(x < y \vee x = y \vee y < x)$

Vogliamo mostrare che **MA** è abbastanza forte da rispecchiare il comportamento delle funzioni calcolabili. Abbiamo bisogno della nozione seguente, che è l'analogo della definibilità in $\mathbf{N}$ per la teoria formale **MA**.

Definizione 1.1 (Rappresentabilità di una funzione in una teoria)**.** Una funzione $\varphi : \mathbf{N}^k \to \mathbf{N}$ è rappresentabile in una teoria T se esiste una formula $F_\varphi(x_1, \ldots, x_k, z)$ nel linguaggio di T tale che

$$T \vdash (\forall x_1) \ldots (\forall x_k)(\forall y)(\forall z)[F_\varphi(x_1, \ldots, x_k, y) \wedge F_\varphi(x_1, \ldots, x_k, z) \to y = z],$$

e per ogni $n_1, \ldots, n_k, m$ in $\mathbf{N}$

$$\text{Se } \varphi(n_1, \ldots, n_k) = m \text{ allora } T \vdash F_\varphi(\bar{n_1}, \ldots, \bar{n_k}, \bar{m}).$$

(se l'ultima implicazione si inverte parliamo di *rappresentabilità forte*).

La nozione naturale di rappresentabilità in una teoria per relazioni è la seguente:

Definizione 1.2 (Rappresentabilità di una relazione in una teoria)**.** Una relazione $R \subseteq \mathbf{N}^k$ è rappresentabile in una teoria T se esiste una formula $F_R(x_1, \ldots, x_k)$ nel linguaggio di T con k variabili libere tale che

(1) Se $(n_1, \ldots, n_k) \in R$ allora $T \vdash F_R(\bar{n_1}, \ldots, \bar{n_k})$,
(2) Se $(n_1, \ldots, n_k) \notin R$ allora $T \vdash \neg F_R(\bar{n_1}, \ldots, \bar{n_k})$

Per dimostrare che tutte le funzioni e le relazioni calcolabili sono rappresentabili nella teoria **MA** abbiamo bisogno delle proposizioni seguenti. Per alleggerire la notazione omettiamo la barra sui termini di tipo n (numerali) che corrispondono ai numeri naturali nel linguaggio di **MA**.

Proposizione 1.3. *Per ogni* $n \in \mathbf{N}$, $\mathbf{MA} \vdash x < n+1 \to x \leq n$.

Dimostrazione. Ricordiamo che $x < n+1$ abbrevia $\exists z(x + z = n+1 \wedge x \neq (n+1))$. Ragioniamo in **MA** per casi. Se $z = 0$ allora per (Ax 5) $x = n+1$, che contraddice $x \neq n+1$. Dunque $z \neq 0$. Da (Ax 3) segue $\exists w(x + (w+1) = n+1)$. Dunque, per (Ax 6) e (Ax 4), $\exists w(x + w = n)$, che per convenzione è abbreviato come $x \leq n$. □

Proposizione 1.4. *Per ogni* $m, n, p \in \mathbf{N}$

(1) $\mathbf{MA} \vdash x \leq n \to x = 0 \vee \cdots \vee x = n$.
(2) *Se* $m + n = p$ *allora* $\mathbf{MA} \vdash m + n = p$.
(3) *Se* $m \cdot n = p$ *allora* $\mathbf{MA} \vdash m \times n = p$.
(4) *Se* $m \neq n$ *allora* $\mathbf{MA} \vdash \neg(m = n)$.

Dimostrazione. Dimostriamo il punto (1) per induzione su n. Per il caso $n = 0$ dobbiamo dimostrare $\mathbf{MA} \vdash x \leq 0 \to x = 0$. $x \leq 0$ abbrevia $\exists z(x + z = 0)$. Ragioniamo in **MA**, e supponiamo $\exists z(x + z = 0)$. Ragioniamo per casi. Se $z \neq 0$, allora per (Ax 3) ho $\exists w(x + (w+1) = 0)$ e da questo, per (Ax 6) e per l'assunzione $\exists z(x + z = 0)$, segue $\exists w((x + w) + 1 = 0)$. Ma questo contraddice (Ax 2). Dunque $z = 0$, e allora $x + 0 = 0$. Per (Ax 5) segue $x = 0$. Per il caso $n+1$, l'ipotesi di induzione è che $\mathbf{MA} \vdash x \leq n \to x = 0 \vee \cdots \vee x = n$. Ragioniamo in **MA**, e supponiamo $x \leq n+1$. Allora $x < n+1 \vee x = n+1$. Allora anche $x \leq n \vee x = n+1$, per la Proposizione 3.1. Per ipotesi induttiva, da $x \leq n$ segue $x = 0 \vee \cdots \vee x = n$ dunque da $x \leq n \vee x = n+1$ segue $x = 0 \vee \cdots \vee x = n \vee x = n+1$.

Dimostriamo il punto (2) per induzione su n. Per il caso $n = 0$ dobbiamo dimostrare che $\mathbf{MA} \vdash m+0 = m$. Questo segue dall'assioma (Ax 5). Per il caso $n+1$, sia $m + (n+1) = q$. Allora $m + n = p$, dove p è $q - 1$. Per ipotesi induttiva, $\mathbf{MA} \vdash m + n = p$. Allora $\mathbf{MA} \vdash (m + n) + 1 = p + 1$. Per (Ax 6) questo implica $\mathbf{MA} \vdash m + (n + 1) = q$.

Dimostriamo il punto (3) per induzione su n. Per il caso $n = 0$ dobbiamo dimostrare che $\mathbf{MA} \vdash m \times 0 = 0$. Questo segue da (Ax 7). Per il caso $n+1$, sia $m \cdot (n+1) = q$. Allora $m \cdot n = p$, dove $q = p + m$. Per ipotesi induttiva $\mathbf{MA} \vdash m \times n = p$. Allora $\mathbf{MA} \vdash (m \times n) + m = p + m$. Per (Ax 8) segue $\mathbf{MA} \vdash m \times (n+1) = p + m$. Dal punto (2) già dimostrato segue $\mathbf{MA} \vdash p + m = q$ (N.B. il termine chiuso $p + m$ non coincide sintatticamente con il termine chiuso q). □

2. Rappresentabilità nell'Aritmetica Minimale

Dimostriamo il teorema seguente. Essenzialmente la dimostrazione consiste nel verificare che gli argomenti usati per dimostrare il Teorema di Rappresentabilità in $\mathbf{N}$ possono essere svolti in base agli assiomi della teoria $\mathbf{MA}$.

Teorema 2.1 (Rappresentabilità in $\mathbf{MA}$)**.** *Le funzioni computabili sono rappresentabili in* $\mathbf{MA}$.

Dimostrazione. **(Extra)** La dimostrazione ripercorre lo schema del Teorema di Rappresentabilità delle funzioni computabili in $\mathbf{N}$, sostituendo argomenti semantici ($\mathbf{N} \models \dots$) con argomenti sintattici ($\mathbf{MA} \vdash \dots$).

Dimostriamo il teorema in tre passi:

- Le funzioni di base sono rappresentabili.
- Le funzioni rappresentabili sono chiuse sotto sostituzione.
- Le funzioni rappresentabili sono chiuse sotto minimo.

Claim 2.2 (Rappresentabilità Funzioni di Base)**.** *Le funzioni di base sono rappresentabili in* $\mathbf{MA}$.

Dimostrazione. L'addizione è rappresentata dalla formula $x+y = z$, la moltiplicazione dalla formula $x \times y = z$, le proiezioni dalle formule $x_i = z$, l'uguaglianza dalla formula $(x = y \wedge z = 1) \vee (x \neq y \wedge z = 0)$. □

Claim 2.3 (Chiusura per Composizione)**.** *Le funzioni rappresentabili in* $\mathbf{MA}$ *sono chiuse per composizione.*

Dimostrazione. Sia $G_i(\vec{x}, y_i)$ il rappresentante di θ_i, $i \in [1, m]$, sia $H(y_1, \dots, y_m, z)$ il rappresentante di ψ. Allora

$$\exists y_1 \dots \exists y_m (G_1(\vec{x}, y_1) \wedge \dots \wedge G_m(\vec{x}, y_m) \wedge H(y_1, \dots, y_m, z))$$

rappresenta $\varphi(\vec{x}) = \psi(\theta_1(\vec{x}), \dots, \theta_m(\vec{x}))$.

Se $\varphi(\vec{k}) = p$ allora esistono $q_1, \dots, q_m \in \mathbf{N}$ tali che $\theta_i(\vec{k}) = q_i$ e $\psi(q_1, \dots, q_m) = p$. Dunque $\mathbf{MA} \vdash G_i(\vec{k}, q_i)$ e $\mathbf{MA} \vdash H(q_1, \dots, q_m, p)$ per ipotesi su G_i e H. □

Claim 2.4 (Chiusura per Minimo)**.** *Le funzioni rappresentabili in* $\mathbf{MA}$ *sono chiuse per minimo.*

Dimostrazione. Sia $G(\vec{x}, z, y)$ un rappresentante di ψ. Allora

$$\forall w (w \leq z \to \exists y (G(\vec{x}, w, y) \wedge (y = 0 \leftrightarrow w = z))$$

rappresenta $\varphi(\vec{x}) = \min z(\psi(\vec{x}, z) = 0)$.

Dimostriamo prima l'univocità di $F(\vec{x}, z)$. Osserviamo che $\mathbf{MA} \vdash F(\vec{x}, z) \to G(\vec{x}, z, 0)$ e che dalla definizione di F segue

$$\mathbf{MA} \vdash (w < z \wedge F(\vec{x}, z)) \to \neg G(\vec{x}, w, 0).$$

Dunque

$$\mathbf{MA} \vdash (w < z \wedge F(\vec{x}, z)) \to \neg F(\vec{x}, w).$$

Dall'Assioma di totalità (Ax 9) dell'ordine $<$ segue

$$\mathbf{MA} \vdash F(\vec{x}, z) \wedge F(\vec{x}, w) \to z = w.$$

Infatti, se $w < z$, dato che $F(\vec{x}, z)$, segue $\neg G(\vec{x}, w, 0)$. D'altra parte, da $F(\vec{x}, w)$ segue $G(\vec{x}, w, 0)$. Dunque $\neg(w < z)$. Se invece $z < w$, dato che $F(\vec{x}, w)$, segue $\neg G(\vec{x}, z, 0)$, ma dato che $F(\vec{x}, z)$, segue $G(\vec{x}, z, 0)$. Dunque $\neg(z < w)$. Per (Ax 9) l'ultima possibilità rimasta è $z = w$.

Dimostriamo ora che $F(\vec{x}, z)$ rispecchia in **MA** il comportamento input-output di $\varphi(\vec{x})$. Siano $k_1, \ldots, k_n, p$ in $\mathbf{N}$ tali che $\varphi(k_1, \ldots, k_n) = p$. Allora, per definizione di φ, abbiamo

- $\psi(\vec{k}, p) = 0$ e,
- per ogni $q < p$ esiste $m \neq 0$ tale che $\psi(\vec{k}, q) = m$.

Per ipotesi su ψ, vale

$$\mathbf{MA} \vdash G(\vec{k}, p, 0),$$

e, per ogni $q < p$ esiste un $m \neq 0$ tale che

$$\mathbf{MA} \vdash G(\vec{k}, q, m).$$

Inoltre per un tale m vale

$$\mathbf{MA} \vdash m \neq 0,$$

perché $m \neq 0$.

Per l'univocità di G vale, per ogni $q < p$,

$$\mathbf{MA} \vdash \neg G(\vec{k}, q, 0).$$

Vogliamo dimostrare che vale $F(\vec{k}, p)$. Abbiamo già visto che $\mathbf{MA} \vdash G(\vec{k}, p, 0)$ e dunque per $w = p$ l'implicazione in F è vera. Consideriamo ora i valori $w < p$. Ragioniamo per casi.

Se $p = 0$ allora $w < p$ contraddice gli assiomi. Dunque

$$\forall w(w < 0 \to \neg G(\vec{k}, w, 0)).$$

Se $p = s + 1$ allora (per la Proposizione 3.2 parte (1)) vale

$$\mathbf{MA} \vdash w < p \to w = 0 \vee \cdots \vee w = s.$$

Dunque anche

$$\mathbf{MA} \vdash \forall w(w < p \to \neg G(\vec{k}, p, 0)).$$

Possiamo allora concludere che

$$\mathbf{MA} \vdash F(\vec{k}, p).$$

□

La dimostrazione del Teorema è così conclusa. □

Lemma 2.5. *Se la funzione caratteristica di una relazione R è rappresentabile in* **MA** *allora R è rappresentabile in* **MA**.

Dimostrazione. Denotiamo con c_R la funzione caratteristica di R, ossia la funzione di tipo $\mathbf{N} \times \mathbf{N} \to \{0, 1\}$ tale che $c_R(a, b) = 1$ se $(a, b) \in R$ e $c_R(a, b) = 0$ se $(a, b) \notin R$. R è calcolabile sse c_R è calcolabile (per definizione). Se c_R è calcolabile è rappresentabile in **MA**. Sia $F(x, y, z)$ la formula che rapprensenta c_R. Per definizione di rapprensetabilità abbiamo, per ogni $a, b \in \mathbf{N}$

(1) Se $c_R(a, b) = 1$ allora $\mathbf{MA} \vdash F(a, b, 1)$
(2) Se $c_R(a, b) = 0$ allora $\mathbf{MA} \vdash F(a, b, 0)$.

Inoltre **MA** dimostra che F è funzionale e che $0 \neq 1$. Dunque abbiamo

(1) Se $c_R(a, b) = 1$ allora $\mathbf{MA} \vdash F(a, b, 1)$
(2) Se $c_R(a, b) = 0$ allora $\mathbf{MA} \vdash \neg F(a, b, 1)$.

Ricapitolando:

(1) Se $(a, b) \in R$ allora $\mathbf{MA} \vdash F(a, b, 1)$
(2) Se $(a, b) \notin R$ allora $\mathbf{MA} \vdash \neg F(a, b, 1)$.

In altre parole, la formula $F(x, y, 1)$ rappresenta in **MA** la relazione R. □

Corollario 2.6. *Tutte le relazioni calcolabili sono rappresentabili in* **MA**.

3. Primo Teorema di Incompletezza di Gödel

Sia T una teoria *formale*, ossia tale che gli elementi di T (assiomi) formano un insieme calcolabile/decidibile. Si dimostra allora facilmente che la seguente relazione binaria $R \subseteq \mathbf{N} \times \mathbf{N}$ è calcolabile:

> "m è il codice di una formula A con una unica variabile libera e n è il codice di una dimostrazione in T della formula ottenuta sostituendo $\bar{m}$ in A all'unica variabile libera di A".

Sia T una teoria formale che estende **MA**.

Sia $F(x, y)$ una formula che *esprime* in T la relazione $R(x, y)$, nel senso seguente: per ogni $a, b \in \mathbf{N}$

$$R(a, b) \Rightarrow T \vdash F(a, b),$$

$$\neg R(a, b) \Rightarrow T \vdash \neg F(a, b).$$

L'esistenza di una tale formula segue dal teorema di rappresentabilità.

Gödel non ha dimostrato il suo Teorema di Incompletezza nella formulazione proposta sopra. In particolare, non ha usato la sola ipotesi di coerenza. Per un passo della dimostrazione ha utilizzato una nozione leggermente più forte, detta ω-coerenza.

Definizione 3.1 (ω-coerenza)**.** Diciamo che una teoria T è ω-coerente se non esiste una formula $A(x)$ tale che

$$\text{Per ogni } n \in \mathbf{N} \quad T \vdash A(n),$$

e

$$T \vdash \neg(\forall x) A(x).$$

Osservazione 3.2. Si osserva che la ω-coerenza implica la coerenza (esercizio).

Consideriamo la formula seguente.

$$(\forall y) \neg F(x, y).$$

Sia m il codice di questa formula. Consideriamo l'enunciato G seguente.

$$\text{(Enunciato G)} \qquad (\forall y) \neg F(m, y).$$

Teorema 3.3 (Primo Teorema di Incompletezza)**.** *Sia T una teoria formale che estende* **MA**.

(1) *Se T è coerente allora G è indimostrabile in T.*

(2) *Se T è ω-coerente allora $\neg G$ è indimostrabile in T.*

Dunque, se T è ω-coerente, allora T è incompleta.

Dimostrazione. Sia T coerente e supponiamo che T dimostri G. Sia d il codice di una dimostrazione. Allora $R(m, d)$. Allora $T \vdash F(m, d)$. Dunque $T \vdash \exists y F(m, y)$. Dunque T è incoerente.

Sia T ω-coerente e supponiamo che T dimostri $\neg G$, i.e., $\exists y F(m, y)$. Per coerenza di T, T non dimostra G. Allora, per ogni numero n, n non è il codice di una dimostrazione di G in T, i.e., per ogni n, non vale $R(m, n)$. Dunque per ogni n, $T \vdash \neg F(m, n)$. Ma allora T è ω-incoerente. □

In altre parole, se T è un'estensione formale di **MA**, o è ω-incoerente o è incompleta.

Osservazione 3.4. Si può dimostrare il Teorema di Gödel sotto la sola ipotesi di coerenza. Questo risultato è dovuto a Rosser.

Sia $H(x, y)$ la formula che rappresenta in T la relazione $S \subseteq \mathbf{N} \times \mathbf{N}$ tale che $S(m, n)$ se e solo se:

> m è il codice di una formula A con una unica variabile libera e n è il codice di una dimostrazione in T della formula ottenuta sostituendo m in $\neg A$ all'unica variabile libera di A.

Consideriamo il seguente enunciato.

$$(\forall y)(F(x,y) \to (\exists z)(z \leq y \wedge H(x,z))).$$

Sia m il codice di questo enunciato. Consideriamo l'enunciato E seguente.

$$\text{(Enunciato E)} \quad (\forall y)(F(m,y) \to (\exists z)(z \leq y \wedge H(m,z))).$$

La lettura intuitiva dell'enunciato E è la seguente: se io sono dimostrabile da una derivazione con codice y allora la mia negazione è dimostrabile da una dimostrazione con codice $z < y$.

Si può dimostrare il seguete Teorema.

Teorema 3.5 (Teorema di Gödel-Rosser)**.** *Sia T una teoria formale che estende* **MA**. *Se T è coerente allora T è incompleta e l'enunciato E non è né dimostrabile né refutabile. Ossia: se T è coerente, allora T è incompleta.*

Dimostrazione. (Esercizio) □

Si osserva che come corollario possiamo ottenere l'indecidibilità della verità aritmetica.

Corollario 3.6 (Indecidibilità della Verità Aritmetica)**.** *La teoria $Th(\mathbf{N}) = \{E \ : \ \mathbf{N} \models E\}$ è algoritmicamente indecidibile.*

Dimostrazione. Dal I Teorema di Gödel sappiamo che se T è una teoria formale aritmetica coerente e completa che estende **MA**, allora l'insieme dei teoremi di T non è decidibile. Ovviamete $Th(\mathbf{N})$ estende **MA** e ovviamente è completa, essendo la teoria di un unico modello, $\mathbf{N}$. Dato che ha un modello, è anche coerente. Dunque l'insieme dei suoi teoremi non è calcolabile (i.e. è algoritmicamente indecidibile). Si osserva facilmente che l'insieme dei teoremi di $Th(\mathbf{N})$ coincide con $Th(\mathbf{N})$ stessa. □

4. Indecibilità algoritmica dell'aritmetica formale

Mostriamo ora come ottenere l'indecidibilità algoritmica dell'aritmetica formale e un risultato sulla complessità della verità aritmetica (Teorema di Tarski) nel contesto dell'approccio originale di Gödel.

Si noti che per i risultati di questo paragrafo **non** si richiede che T sia una teoria formale.

Il seguente risultato e la sua dimostrazione sono la riformulazione del Teorema di Tarski sull'indecidibilità di $Th(\mathbf{N})$, dove la nozione di definibilità in $\mathbf{N}$ viene sostituita dalla nozione di definibilità/rappresentabilità in una teoria T.

Teorema 4.1. *Se la funzione diagonale δ è rappresentabile in T e T è coerente allora i teoremi di T non sono esprimibili in T.*

Dimostrazione. Sia D rappresentabile in T. Sia $D(x,y)$ la formula che la rappresenta. Allora vale

$$D(k) = j \Rightarrow T \vdash D(k,j)$$

e, per ogni $k \in \mathbf{N}$,

$$T \vdash \forall y \forall z((D(k,y) \wedge D(k,z)) \to y = z).$$

Supponiamo per assurdo che i teoremi di T siano esprimibili in T. Sia $V(y)$ la formula che li esprime. Allora vale, per ogni enunciato H,

$$T \vdash H \Rightarrow T \vdash V(cod(H))$$

e

$$T \nvdash H \Rightarrow T \vdash \neg V(cod(H)),$$

dove indichiamo con $cod(H)$ il codice dell'enunciato H.

Consideriamo la formula $A(x)$ seguente.

$$\forall y(D(x,y) \to \neg V(y)).$$

Sia p il codice di questa formula. Consideriamo $A(p)$.

$$\forall y(D(p,y) \to \neg V(y)).$$

Sia q il codice di $A(p)$. Per definizione di δ abbiamo $\delta(p) = q$. Dunque

$$T \vdash D(p, q).$$

Vogliamo ora dimostrare che $T \vdash \neg V(q)$. Ragioniamo per casi.

(Caso 1) Se $T \nvdash A(p)$, allora $A(p)$ non è un teorema di T e dunque $code(A(p)) = q \notin Teor(T)$. Dunque, per esprimibilità,

$$T \vdash \neg V(q).$$

(Caso 2) Se $T \vdash A(p)$, allora anche

$$T \vdash D(p, q) \to \neg V(q).$$

Dunque

$$T \vdash \neg V(q).$$

Abbiamo dimostrato così che $T \vdash \neg V(q)$.

Da $T \vdash D(p, q)$ e dall'univocità della formula D segue che

$$T \vdash D(p, y) \to y = q.$$

Inoltre vale (dato che $T \vdash V(q)$)

$$T \vdash y = q \to \neg V(y).$$

Dunque

$$T \vdash D(p, y) \to \neg V(y).$$

Per generalizzazione abbiamo allora

$$T \vdash \forall y(D(p, y) \to \neg V(y)),$$

ossia

$$T \vdash A(p).$$

Dunque $T \vdash V(code(A(p))$, ossia

$$T \vdash V(q).$$

Dunque T è incoerente. Contraddizione.

□

Corollario 4.2. *Se T è coerente e tutte le funzioni calcolabili sono rappresentabili in T allora i teoremi di T non sono esprimibili in T. In particolare non sono un insieme calcolabile.*

Diciamo che una teoria T è indecidibile se l'insieme dei suoi teoremi non è calcolabile.

Corollario 4.3. *Se* **MA** *è coerente allora ogni estensione coerente di* **MA** *è indecidibile (ossia l'insieme dei suoi teoremi è indecidibile).*

Dimostrazione. Se T è un estensione coerente di **MA**, allora ogni funzione calcolabile è rappresentabile in T. □

In particolare la verità aritmetica $Th(\mathbf{N})$ non è decidibile da un algoritmo, fatto che abbiamo già dimostrato.

Un altro corollario di rilievo riguarda l'impossibilità di risolvere con un algoritmo il Problema della Decisione della validità logica, almeno per il linguaggio dell'aritmetica formale e sotto l'ipotesi che **MA** sia coerente. Supponiamo infatti di avere un algoritmo che risponde alla domanda: l'enunciato E (nel lingugaggio di **MA**) è una verità logica? Potremmo allora rispondere alla domanda: l'enunciato F è un teorema di **MA**? Infatti, dato che **MA** è finitamente assiomatizzata, vale $\mathbf{MA} \models F$ se e solo se $\models A \to F$, dove A è la congiunzione di tutti gli assiomi di **MA**. Ma sappiamo che i teoremi di **MA** sono indecidibli, se **MA** è coerente.

Corollario 4.4 (Teorema di Church - Indecidibilità del Problema della Decisione (forma debole))**.** *Se* **MA** *è coerente, il problema della verità logica per enunciati nel linguaggio di* **MA** *è indecidibile algoritmicamente.*

Si tratta di una versione debole del Teorema di Church, che sancisce l'indecidibilità algoritmica della verità logica per una classe di linguaggi più ampia e senza ipotesi aggiuntive.

LOGICA MATEMATICA
ESERICIZI DI LOGICA PREDICATIVA

1. Routine

Esercizio 1.1. Dimostrare i seguenti punti, dove $F \equiv G$ abbrevia $\models F \leftrightarrow G$:

(1) $\exists x(F \vee G) \equiv \exists xF \vee \exists xG$
(2) $\forall x(F \wedge G) \equiv \forall xF \wedge \exists xG$
(3) $\neg\exists x\neg F \equiv \forall xF$
(4) $\neg\forall x\neg F \equiv x\exists F$
(5) $\exists x\exists yF \equiv \exists y\exists xF$, $\forall x\forall yF \equiv \forall y\forall xF$
(6) Se x non occorre in F,
   (a) $F \vee \exists xG \equiv \exists x(F \vee G)$,
   (b) $F \wedge \exists xG \equiv \exists x(F \wedge G)$
   (c) $F \wedge \forall xG \equiv \forall x(F \wedge G)$,
   (d) $F \vee \forall xG \equiv \forall x(F \vee G)$

Esercizio 1.2. I seguenti i enunciati sono validi, insoddisfacibili o soddisfacibili (ma non validi)?

(1) $(\exists xP(x) \leftrightarrow \exists xQ(x)) \rightarrow \forall x(P(x) \leftrightarrow Q(x))$;
(2) $(\forall x\exists yA(x,y) \rightarrow \forall zB(z)) \rightarrow \forall zB(z)$.
(3) $(\exists xP(x) \wedge \forall y\neg P(y))$
(4) $\forall x\exists yR(x,y) \rightarrow \exists y\forall xR(x,y)$
(5) $\exists y\forall xR(x,y) \rightarrow \forall x\exists yR(x,y)$
(6) $\exists x\forall yR(x,y)$
(7) $\exists xP(x) \rightarrow \exists x\forall yP(x)$

Esercizio 1.3. Formalizzare le proposizioni seguenti con enunciati nel linguaggio predicativo $\mathcal{L}$ composto da un solo simbolo $\in$ (x,y) di relazione binaria il cui significato intuitivo è $x \in y$. Per esempio in questo linguaggio, *Esiste l'insieme vuoto* si esprime con l'enunciato $\exists x\forall y\neg(y \in x)$ (esiste un insieme, x tale che nessun y è elemento di x.

(1) Ogni insieme x è sottinsieme di un qualche insieme y.
(2) Per ogni insieme x esiste l'insieme y complemento di x.
(3) Per ogni coppia di insiemi x e y esiste l'insieme intersezione $x \cap y$.
(4) Esiste un insieme che è sottinsieme di tutti gli insiemi.

Esercizio 1.4. Formalizzare le seguenti proposizioni nel linguaggio composto da un singolo simbolo di predicato a due posti $E(x,y)$ (da leggersi intuitivamente come *esiste un arco tra i vertici* x e y).

(1) Esiste un vertice isolato (ossia non connesso da un arco a nessun altro vertice).
(2) I vertici sono tutti connessi tra di loro (i.e. il grafo è *completo*).
(3) Non esistono archi tra un vertice e se stesso.

Esercizio 1.5. Formalizzare le seguenti proposizioni nel linguaggio composto da un singolo simbolo di predicato a due posti $R(x,y)$ e un simbolo di predicato a due posti $U(x,y)$ (da leggersi come l'identità).

(1) R è un ordine.
(2) R è un ordine totale.
(3) R ha un elemento minimo.
(4) R non ha un massimo.

Note preparate da Lorenzo Carlucci, lorenzo.carlucci@uniroma1.it. Alcuni degli esercizi proposti sono tratti dai manuali di E. Mendelson, *Introduction to Mathematical Logic*, e di D. Van Dalen, *Logic and Structure*.

Esercizio 1.6. Tradurre le seguenti proposizioni in linguaggio naturale interpretando $P(x)$ com x è un punto, $S(x)$ come x è una retta, $L(x,y)$ come x giace su y, $C(x,y,z)$ come x,y,z sono collineari e $I(x,y)$ come $x=y$.

(1) $\forall x(P(x) \to \neg S(x))$
(2) $\forall x \forall y(L(x,y) \to (P(x) \wedge S(y)))$
(3) $\forall x(S(x) \to \exists y \exists z(\neg I(y,z) \wedge L(y,x) \wedge L(z,x)))$
(4) $\forall x \forall y((P(x) \wedge P(y) \wedge \neg I(x,y)) \to (\exists z)(S(z) \wedge L(x,z) \wedge L(y,z)))$
(5) $\forall x \forall y \forall z \forall w((P(x) \wedge P(y) \wedge \neg I(x,y) \wedge S(z) \wedge S(w) \wedge L(x,z) \wedge L(y,z) \wedge L(x,w) \wedge L(y,w)) \to I(z,w))$
(6) $\exists x \exists y \exists z(P(x) \wedge P(y) \wedge P(z) \wedge \neg C(x,y,z))$

Esercizio 1.7. Usando il linguaggio dell'esercizio precedente tranne il predicato $C(x,y,z)$ scrivere una formula $F(x,y,z)$ che esprima il fatto che x,y,z sono collineari.

Esercizio 1.8. Tradurre in linguaggio naturale il seguente enunciato (nel linguaggio formale dell'esercizio precedente):

$$\forall u \forall v((S(u) \wedge S(v) \wedge \neg I(u,v)) \to \forall x \forall y((L(x,u) \wedge L(x,v) \wedge L(y,u) \wedge L(y,v)) \to I(x,y))$$

Esercizio 1.9. Consideriamo il linguaggio composto da una costante c, da un simbolo di relazione a due posti $S(x,y)$ e da un simbolo di relazione a tre posti $R(x,y,z)$. Per ognuno degli enunciati seguenti descrivere una interpretazione in cui l'enunciato è vero e una in cui è falso.

(1) $\forall x \exists y \forall z(S(x,c) \to R(x,y,z))$.
(2) $\exists y \forall x \forall z(S(x,c) \to R(x,y,z))$.
(3) $\forall x \forall y(S(x,y) \to S(y,x))$.

Esercizio 1.10. Consideriamo il linguaggio composto da un unico simbolo di relazione $R(x,y)$ a due posti. Scrivere un enunciato vero nell'interpretazione $\mathfrak{A}_1$ con dominio $\mathbf{N}$ e $\leq$ come interpretazione di R ma falso nell'interpretazione $\mathfrak{A}_2$ con dominio $\mathbf{Q}$ e $\leq$ come interpretazione di R. Stessa cosa rispetto a $\mathfrak{A}_3$ con dominio $\mathbb{Z}$ e R interpretata sempre con $\leq$.

Esercizio 1.11. Consideriamo un linguaggio $\mathcal{L}$ contenente, tra l'altro, un simbolo di predicato a due posti $I(x,y)$ da interpretare come l'identità tra x e y. Tradurre in $\mathcal{L}$ le seguenti proposizioni:

(1) Maria ha avuto almeno tre mariti.
(2) Al massimo due persone in questa festa conoscono tutti.

Esercizio 1.12. Consideriamo un linguaggio $\mathcal{L}$ contenente, tra l'altro, un simbolo di predicato a due posti $I(x,y)$ da interpretare come l'identità tra x e y. Sia $F(x)$ una formula in questo linguaggio, contenente x come sola variabli libera (i.e., non quantificata). Scrivere, usando $F(x)$ e il predicato I una formula che esprima il fatto che esiste *un unico* x tale che $F(x)$.

Esercizio 1.13. Scrivere, per ogni $n \in \mathbf{N}$, un enunciato E_n che esprima: *esistono almeno n elementi distinti*. Scrivere un enunciato che esprima *eistono esattamente n elementi distinti*.

Esercizio 1.14. Formalizzare le seguenti proposizioni in linguaggio predicativo.

(1) Qualche uomo è un genio. Nessuna scimmia è un uomo. Qualche genio non è una scimmia.
(2) Qualche uomo è un genio. Nessuna scimmia è un uomo. Nessuna scimmia è un genio.
(3) Qualche genio è scapolo. Qualche studente non è scapolo. Qualche studente non è un genio.
(4) Per ogni numero x,y,z se $x > y$ e $y > z$ allora $x > z$. Per ogni x, $x > x$ è falso. (Dunque) per ogni x,y se $x > y$ allora non vale $y > x$.
(5) Nessuno studente di Fisica è più intelligente di tutti gli studenti di Matematica. (Dunque) qualche studente di Matematica è più intelligente di ogni studente di Fisica.
(6) Per ogni insieme x esiste un insieme y di cardinalità strettamente maggiore. Se x è un sottinsieme di y allora la cardinalità di x non è strettamente maggiore di quella di y. Ogni insieme è un sottinsieme dell'insieme V. (Dunque) V non è un insieme.
(7) Per ogni intero positivo x vale $x \leq x$. Per ogni intero positivo x,y,z se $x \leq y$ e $y \leq z$ allora $x \leq z$. Per ogni intero positivo x,y vale $x \leq y$ oppure $y \leq x$. (Dunque) esiste un intero positivo y tale che per ogni intero positivo x si ha $y \leq x$.

Esercizio 1.15. Trovare controesempi alla validità dei seguenti enunciati.

(1) $(\forall x\forall y\forall z(A(x,y)\wedge A(y,z)\to A(x,z))\wedge\forall x\neg A(x,x))\to\exists x\forall y\neg A(x,y)$.
(2) $\forall x\exists yA(x,y)\to\exists yA(y,y)$
(3) $\exists x\exists yA(x,y)\to\exists yA(y,y)$
(4) $(\exists xP(x)\leftrightarrow\exists xQ(x))\to\forall x(P(x)\leftrightarrow Q(x))$.
(5) $(\forall x\forall y(R(x,y)\to R(y,x)\wedge\forall x\forall y\forall z(R(x,y)\wedge R(y,z)\to R(x,z))\to\forall xR(x,x)))$.
(6) $\forall x\forall y\forall z(R(x,x)\wedge(R(x,z)\to R(x,y)\vee R(y,z)))\to\exists y\forall zR(y,z)$.

Esercizio 1.16. Decidere se i seguenti enunciati sono validi o non validi.

(1) $(\forall x(A(x)\vee B(x))\to(\forall xA(x)\vee\forall xB(x))$
(2) $(\exists xA(x)\wedge\exists xB(x))\to\exists x(A(x)\wedge B(x))$
(3) $(\forall xA(x)\to\forall xB(x))\to\forall x(A(x)\to B(x))$
(4) $\forall x(A(x)\to B(x))\to(\forall xA(x)\to\forall xB(x))$
(5) $(\forall xA(x)\to\exists xB(x))\leftrightarrow\exists x(A(x)\to B(x))$
(6) $(\exists xA(x)\to\forall xB(x))\to\forall x(A(x)\to B(x))$

Esercizio 1.17. Siano A e B formule (non necessariamente enunciati). Supponiamo che per ogni struttura $\mathcal{A}$ si ha che se $\mathcal{A}\models B$ allora $\mathcal{A}\models C$. Dimostrare che allora vale anche che se $\models B$ allora $\models C$. Dimostrare. Dimostrare anche che il viceversa non vale.

Esercizio 1.18. Scrivere enunciati che distinguono tra le seguenti strutture: $(\mathbb{N},<)$, $(\mathbb{Q},<)$, $(\mathbb{Z},<)$ usando il linguaggio del'aritmetica seguente: $\{=,<,+,*,0,1\}$.

Esercizio 1.19. Sia $\mathcal{L}$ il linguaggio contenente i simboli $=$ (simbolo speciale), $+$, $*$ (funzioni a due argomenti), S (funzione a un argomento) e 0 (costante). Scrivere formule che definiscono i seguenti concetti se interpretare sulla struttura standard con dominio $\mathbb{N}$ e interpretazione standard degli altri simboli (addizione, prodotto, successore, zero).

(1) x è un numero primo.
(2) x e y sono relativamente primi.
(3) x minimo primo maggiore di y.
(4) x massimo tale che $2x<y$.
(5) La Congettura di Goldbach.
(6) Nel linguaggio che estende $\mathcal{L}$ con una funzione exp a due argomenti interpretata com e $n,m\mapsto n^m$ per $n,m\neq 0$ e $0,m\mapsto 0$; esprimere il Teorema di Fermat.

Se $F(x)$ è una formula e t un termine indichiamo con $F(t/x)$ la formula ottenuta sostituendo ogni occorrenza di x in $F(x)$ con t.

Esercizio 1.20. Sia $F(x)$ una formula e t un termine. Se nessuna variabile di t occorre in $F(x)$ allora: per ogni struttura $\mathfrak{A}$ e per ogni α su $\mathfrak{A}$

$$\mathfrak{A}\models F(t/x)[\alpha] \text{ se e solo se } \mathfrak{A}\models F(x)[\alpha\binom{x}{\alpha(t)}].$$

Esercizio 1.21. Se $F(x)$ è una formula e t un termine in cui non compare nessuna variabile di F allora

$$\models\forall xF(x)\to F(t/x).$$

Esercizio 1.22. L'esercizio precedente rimane vero se tutte le variabili di t rimangono libere nella formula $F(t/x)$? E rimane vero senza condizioni sulle variabili di t?

2. Forma Normale Prenessa

Una formula è in *forma normale negativa* se ogni occorrenza del simbolo di negazione occorre soltanto davanti a formule atomiche.

Osservazione 2.1 (Sostituire sottoformule equivalenti)**.** Se G è una sottoformula di F e G è equivalente a G' allora F è equivalente alla formula ottenuta da F sostituendo ogni occorrenza di G con G'.

Osservazione 2.2 (Rinominare variabili vincolate)**.** Se y non appare in $\forall x G$, allora $\forall y G(y)$ è equivalente a $\forall x G$, dove $G(y)$ indica la formula ottenuta da G sostituendo tutte le occorrenze di x con y.

Esercizio 2.3. Dimostrare che per ogni linguaggio predicativo $\mathcal{L}$ ogni formula di $\mathcal{L}$ è logicamente equivalente a una formula in forma normale prenessa. (Suggerimento: procedere per induzione sulle formule, usando le osservazioni precedenti e le equivalenze logiche del primo esercizio).

3. Identità

Per un simbolo binario di relazione I chiamiamo *Assiomi dell'Identità* per I gli enunciati seguenti.

(1) $\forall x I(x,x)$.
(2) $\forall x \forall y (I(x,y) \to (F(x,x) \to F(x,y)))$, per ogni formula $F(x,x)$, dove $F(x,y)$ è una formula ottenuta da $F(x,x)$ per sostituzione di alcune (non necessariamente tutte) le occorrenze di x con y, e y è libera per x in $F(x,x)$.

Lemma 3.1 (Esercizio)**.** *Se T è una teoria che implica gli Assiomi dell'Identità per un simbolo binario di relazione I, allora per ogni termine t, s, r, valgono i seguenti punti.*

(1) $T \models I(t,t)$.
(2) $T \models I(t,s) \to I(s,t)$.
(3) $T \models I(t,s) \to (I(s,r) \to I(t,r))$.

Corollario 3.2 (Esercizio)**.** *Sia $\mathfrak{A}$ un modello di una teoria T che implica gli assiomi dell'identità per una relazione binaria I. La relazione $I^{\mathfrak{A}}$ è una relazione di equivalenza su A.*

Proposizione 3.3. *Sia T una teoria che implica gli assiomi dell'identità per una relazione binaria I. Se T ha un modello allora T ha un modello in cui I è interpretata come l'identità. Chiamiamo un tale modello modello normale.*

Sia $\mathfrak{A}$ un modello di T. Allora $I^{\mathfrak{A}}$ è una relazione di equivalenza sul dominio A. Per leggibilità denotiamo questa relazione con E. Quozientiamo A rispetto a E. Poniamo

$$B = \{[a]_E \quad a \in A\},$$

dove con $[a]_E$ indichiamo la classe di equivalenza dell'elemento a. Definiamo l'interpretazione in $\mathfrak{B}$ delle costanti, relazioni e funzioni del linguaggio.

(Costanti) Poniamo $c_i^{\mathfrak{B}}$ uguale a $[c_i^{\mathfrak{A}}]_E$.
(Relazioni) $(b_1, \ldots, b_k) \in R^{\mathfrak{B}}$ se e solo se $(a_1, \ldots, a_k) \in R^{\mathfrak{A}}$, dove a_i è un rappresentante della classe di equivalenza b_i. Per dimostrare che la definizione è ben posta (e che non dipende dalla scelta dei rappresentanti) occorre osservare che, per ogni simbolo di relazione R,

$$T \models \bigwedge_{i=1}^{k} I(x_i, y_i) \to (R(x_1, \ldots, x_k) \leftrightarrow R(y_1, \ldots, y_k)$$

segue dal fatto che T implica gli Assiomi dell'Identità per I.
(Funzioni) $f^{\mathfrak{B}}(b_1, \ldots, b_k)$ è definito come $[f^{\mathfrak{A}}(a_1, \ldots, a_k)]_E$, dove a_i è un rappresentante della classe di equivalenza b_i. Per dimostrare che la definizione è ben posta (e che non dipende dalla scelta dei rappresentanti) occorre osservare che, per ogni simbolo di funzione f,

$$T \models \bigwedge_{i=1}^{k} x_i = y_i \to I(f(x_1, \ldots, x_k), f(y_1, \ldots, y_k))$$

segue dal fatto che T implica gli assiomi dell'identità per I.

Esercizio 3.4. Dimostrare che $I^{\mathfrak{B}}$ è l'identità su B.

Esercizio 3.5. Dimostrare che $\mathfrak{B}$ è un modello di T.

Sia $\mathfrak{B}$ una struttura per $\mathcal{L}$ e $A \subseteq B$. Diciamo che A è l'universo di una sottostruttura di $\mathfrak{B}$ se esiste una struttura $\mathfrak{A}$ sottostruttura di $\mathfrak{B}$ con dominio A.

Esercizio 4.1. Se $\mathcal{L}$ contiene solo simboli di relazione e $\mathfrak{B}$ è una struttura per $\mathcal{L}$ allora ogni sottinsieme di B è universo di una sottostruttura di $\mathfrak{B}$.

Esercizio 4.2. Sia $\mathfrak{B}$ una struttura adeguata per $\mathcal{L}$ e sia $A \subseteq B$. Allora esiste al più una sottostruttura di $\mathfrak{B}$ con universo A.

Esercizio 4.3. $A \subseteq B$ è universo di una sottostruttura di $\mathfrak{B}$ se e solo se valgono i seguenti due punti:

(1) Per ogni simbolo di costante c, $c^{\mathfrak{B}} \in A$ (diciamo che A *contiene le costanti* di $\mathfrak{B}$), e
(2) Per ogni simbolo di funzione f a n argomenti, per ogni $a_1, \ldots, a_n \in A$, $f^{\mathfrak{B}}(a_1, \ldots, a_n) \in A$ (diciamo che A *è chiuso sotto le funzioni di* $\mathfrak{B}$).

Esercizio 4.4. Sia $\{\mathfrak{A}_i : i \in I\}$ una famiglia (con I di cardinalità arbitraria) di sottostrutture di una struttura $\mathfrak{B}$. Allora $\bigcap_{i\in I} \mathfrak{A}_i$ è il dominio di una sottostruttura di $\mathfrak{B}$.

Esercizio 4.5. Sia $\mathfrak{A} = (\mathbf{N}, <)$ e sia $\mathfrak{B} = (\mathbf{N} - \{0\}, <)$. Dimostrare i seguenti punti:

(1) $\mathfrak{A}$ e $\mathfrak{B}$ sono isomorfe.
(2) $\mathfrak{A}$ e $\mathfrak{B}$ soddisfano gli stessi enunciati.
(3) $\mathfrak{B}$ è una sottostruttura di $\mathfrak{A}$.
(4) $\mathfrak{B}$ non soddisfa la Proprietà del Testimone relativamente ad $\mathfrak{A}$.

Se t è un termine indichiamo con $t(x_1, \ldots, x_n)$ il fatto che tutte le variabili di t sono in $\{x_1, \ldots, x_n\}$. Sia $\mathfrak{A}$ una struttura per $\mathcal{L}$ e $t(x_1, \ldots, x_n)$ un termine in $\mathcal{L}$. Per $a_1, \ldots, a_n \in A$ indichiamo con $t^{\mathfrak{A}}[a_1, \ldots, a_n]$ (interpretazione di t in $\mathfrak{A}$ relativamente a $a_1, \ldots, a_n$) l'elemento di A così definito per ricorsione su t:

(1) Se t è una costante, $t^{\mathfrak{A}}[a_1, \ldots, a_n] = c^{\mathfrak{A}}$.
(2) Se t è una variabile v_i con $i \in \{1, \ldots, n\}$; $t^{\mathfrak{A}}[a_1, \ldots, a_n] = a_i$.
(3) Se t è $f(t_1, \ldots, t_m)$, $t^{\mathfrak{A}}[a_1, \ldots, a_n] = f^{\mathfrak{A}}(t_1^s trA[a_1, \ldots, a_n], \ldots, t_m^{\mathfrak{A}}[a_1, \ldots, a_n])$.

Esercizio 4.6. Se $\mathfrak{A}$ è una sottostruttura di $\mathfrak{B}$ allora per ogni $a_1, \ldots, a_n \in A$,

$$t^{\mathfrak{B}}[a_1, \ldots, a_n] = t^{\mathfrak{A}}[a_1, \ldots, a_n].$$

Una funzione $h : A \to B$ è un omomorfismo tra le strutture $\mathfrak{A}$ e $\mathfrak{B}$ adeguate per $\mathcal{L}$ sse valgono i seguenti punti:

(1) Per ogni simbolo di costante c:
$$h(c^{\mathfrak{A}}) = c^{\mathfrak{B}}.$$
(2) Per ogni simbolo di funzione f a n argomenti, per ogni $a_1, \ldots, a_n \in A$:
$$h(f^{\mathfrak{A}}(a_1, \ldots, a_n)) = f^{\mathfrak{B}}(h(a_1), \ldots, h(a_n))$$
(3) Per ogni simbolo di relazione R a n posti, per ogni $a_1, \ldots, a_n \in A$:
$$\text{Se } (a_1, \ldots, a_n) \in R^{\mathfrak{A}} \text{ allora } (h(a_1), \ldots, h(a_n)) \in R^{\mathfrak{B}}.$$

Si osserva facilmente che l'inclusione insiemistica $A \subseteq B$ è un omormorfismo.

Esercizio 4.7. Se esiste un omomorfismo $h : \mathfrak{A} \to \mathfrak{B}$ allora per ogni $a_1, \ldots, a_n \in A$,

$$t^{\mathfrak{B}}[h(a_1), \ldots, h(a_n)] = h(t^{\mathfrak{A}}[a_1, \ldots, a_n]).$$

Esercizio 4.8. Se $\mathfrak{A}$ è una sottostruttura di $\mathfrak{B}$ allora per ogni formula $F(y_1, \ldots, y_n)$ senza quantificatori, per ogni $a_1, \ldots, a_n \in A$:

$$\mathfrak{B} \models F(y_1, \ldots, y_n)[a_1, \ldots, a_n] \text{ se e solo se } \mathfrak{A} \models F(y_1, \ldots, y_n)[a_1, \ldots, a_n].$$

Esercizio 4.9. Sia $\mathfrak{A}$ sottostruttura di $\mathfrak{B}$. Se $F(y_1, \ldots, y_n)$ è una formula di tipo $\forall x_1 \ldots \forall x_m G(x_1, \ldots, x_m, y_1, \ldots, y_m)$ allora, per ogni $a_1, \ldots, a_n \in A$:

$$\text{Se } \mathfrak{B} \models F(y_1, \ldots, y_n)[a_1, \ldots, a_n] \text{ allora } \mathfrak{A} \models F(y_1, \ldots, y_n)[a_1, \ldots, a_n].$$

Esercizio 4.10. Sia $\mathfrak{A}$ sottostruttura di $\mathfrak{B}$. Se $F(y_1, \ldots, y_n)$ è una formula di tipo $\exists x_1 \ldots \exists x_m G(x_1, \ldots, x_m, y_1, \ldots, y_m)$ allora, per ogni $a_1, \ldots, a_n \in A$:

$$\text{Se } \mathfrak{A} \models F(y_1, \ldots, y_n)[a_1, \ldots, a_n] \text{ allora } \mathfrak{B} \models F(y_1, \ldots, y_n)[a_1, \ldots, a_n].$$

Esercizio 4.11. Se $\mathfrak{A}$ è sottostruttura di $\mathfrak{B}$ dimostrare l'equivalenza tra i due punti seguenti:

(1) $\mathfrak{A}$ soddisfa la Proprietà del Testimone relativamente a $\mathfrak{B}$,
(2) Per ogni formula $F(y_1, \ldots, y_n)$ per ogni $a_1, \ldots, a_n \in A$,

$$\mathfrak{B} \models F(y_1, \ldots, y_n)[a_1, \ldots, a_n] \text{ se e solo se } \mathfrak{A} \models F(y_1, \ldots, y_n)[a_1, \ldots, a_n].$$

Esercizio 4.12. Abbiamo visto a lezione che se $\mathfrak{A}$ soddisfa la Proprietà del Testimone relativamente a $\mathfrak{B}$ allora $\mathfrak{A}$ e $\mathfrak{B}$ soddisfano gli stessi enunciati. Dimostrare con un controesempio che non vale il viceversa.

Diciamo che un insieme $X \subseteq A$ è definibile in una struttura $\mathfrak{A}$ se esiste una formula $F(x, y_1, \ldots, y_n)$ e $(a_1, \ldots, a_n) \in A^n$ (dove $n \geq 0$) tali che

$$X = \{a \in A : \mathfrak{A} \models F(x)[a, a_1, \ldots, a_n]\}.$$

Gli elementi $a_1, \ldots, a_n$ vengono detti parametri. Se $n = 0$ diciamo che X è definibile senza parametri.

Esercizio 4.13. Per ogni intero positivo n l'insieme delle coppie di vertici (u, v) tali che esiste un cammino di lunghezza al più n tra u e v è definibile senza parametri.

Esercizio 4.14. L'insieme dei numeri positivi è definibile in $\mathcal{R} = (\mathbb{R}, 0, 1, +, \times)$ (la struttura standard dei Reali).

Esercizio 4.15. L'insieme dei naturali è definibile in $\mathcal{Z} = (\mathbb{Z}, +, \times)$ (la struttura standard degli interi).

Esercizio 4.16. Gli insiemi definibili sono punti fissi di ogni automorfismo di una struttura $\mathfrak{A}$; ossia: se $f : \mathfrak{A} \to \mathfrak{A}$ è un isomorfismo e $X \subseteq A$ è definibile, allora $f(X) = \{f(x) : x \in X\} = X$.

Esercizio 4.17. L'insieme dei numeri razionali negativi non è definibile in $\mathcal{Q} = (\mathbb{Q}, <)$.

Esercizio 4.18. L'insieme $\{\sqrt{2}, -\sqrt{2}\}$ è definibile in $\mathcal{C} = (\mathbb{C}, 0, 1, +, \times\})$ senza parametri, ossia da una formula $F(x)$ con la sola variabile libera x.

Esercizio 4.19. Scrivere una teoria i cui modelli siano tutti e soli i campi algebricamente chiusi.

Esercizio 4.20. Per ogni n intero positvo definire una teoria T_n in un adeguato linguaggio tale che tutti i modelli di cardinalità n di T_n sono isomorfi.

Esercizio 4.21. Sia κ un cardinale più che numerabile. È vero che tutti i modelli di DLO di cardinalità κ sono isomorfi?

Esercizio 4.22. * Si consideri la teoria T ottenuta aggiungendo agli assiomi di gruppo abeliano l'assioma

$$\forall x \exists y \forall z ((\underbrace{y + \cdots + y}_{n} = x) \leftrightarrow (y = z)),$$

per ogni $n > 1$. Questi enunciati esprimono la divisibilità unica degli elementi del gruppo. La teoria T ha modelli numerabili non-isomorfi. Per ogni cardinalità più che numerabile κ, tutti i modelli di T di cardinalità κ sono isomorfi.

Esercizio 4.23. * Sia $\mathfrak{A} = (\mathbb{R}, <, f)$ con f una funzione unaria. Non esiste un enunciato E (nel linguaggio adeguato) tale che $\mathfrak{A} \models E$ se e solo se per ogni $r \in \mathbb{R}$ $f(r) > 0$.

Esercizio 4.24. Sia $\mathfrak{A} = (A, E)$ con E una relazione di equivalenza su A. Se le classi di equivalenza di E sono tutte infinite e sono in quantità numerabile, allora i modelli numerabili della teoria $Th(\mathfrak{A})$ sono tutti isomorfi.

Esercizio 4.25. Una teoria T è completa se e solo se tutti i suoi modelli sono elementarmente equivalenti (i.e., soddisfano gli stessi enunciati).

5. Completezza

Esercizio 5.1. Siano $T_0 \subseteq T_1 \subseteq T_2 \subseteq \ldots$ teorie. Dimostrare che se ogni T_n è coerente allora $\bigcup_{n\in\mathbf{N}}$ è coerente.

Esercizio 5.2. Completare la dimostrazione del Lemma di Lindenbaum, dimostrando i seguenti punti:

(1) Ogni S_n è coerente.
(2) $\bigcup_{n\in\mathbf{N}} S_n$ è completa.

Esercizio 5.3. Sia T una teoria, c una costante. Sai y una variabile che non compare né in T né in F. Dimostrare che se $T \vdash F$ allora $T[c/y] \vdash F[c/y]$ dove denotiamo con $T[c, y]$ (rispettivamente $F[c/y]$) il risultato di sostituire ogni occorrenza di c negli enunciati di T (rispettivamente in F) con y. La dimostrazione si può condurre per induzione sulla derivazione $T \vdash F$.

Esercizio 5.4. Nella costruzione del modello dei termini $\mathcal{M}$ per una teoria T senza identità abbiamo posto $c^{\mathcal{M}} = c$ per le costanti e $f^{\mathcal{M}} : (t_1, \ldots, t_n) \mapsto f(t_1, \ldots, t_n)$ per le funzioni. Dimostrare per induzione sui termini che per ogni termine t vale $t^{\mathcal{M}} = t$. Dimostrare che se $F(x)$ è una formula con una variabile libera e t è un termine chiuso, nel modello dei termini vale che: $\mathcal{M} \models F(x)[\binom{x}{t}]$ se e solo se $\mathcal{M} \models F[x/t]$.

Esercizio 5.5. Sia $\mathcal{G}$ un qualunque gruppo non banale. Consideriamo $T = Th(\mathcal{G})$. Dimostrare che T non è una teoria con testimoni.

6. Compattezza

Esercizio 6.1. Sia T una teoria che assiomatizza una proprietà P di strutture. Dimostrare che se P è anche finitamente assiomatizzabile allora P è finitamente assiomatizzabile da un sottinsieme di T.

Esercizio 6.2. Siano T_1 e T_2 due teorei in un linguaggio $\mathcal{L}$. Supponiamo che per ogni struttura $\mathfrak{A}$ adeguata per $\mathcal{L}$, $\mathfrak{A} \models T_1$ se e solo se if $\mathfrak{A} \not\models T_2$. Allora T_1 e T_2 sono finitamente assiomatizzabili.

(Suggerimento: Ragionare per assurdo e usare il Teorema di Compattezza per ottenere un modello di una teoria insoddisfacibile).

Esercizio 6.3. Se X è un insieme denotiamo con $[X]^2$ l'insieme delle coppie (non-ordinate) di elementi di X. Una colorazione di $[X]^2$ in 2 colori è una funzione $f : [X]^2 \to \{0, 1\}$.

Consideriamo il seguente Teorema.

Teorema di Ramsey Infinito Sia X un insieme numerabile. Per ogni colorazione f di $[X]^2$ in 2 colori esiste un H sottinsieme infinito di X tale che tutte le coppie prese in H hanno lo stesso colore; ossia f ristretta a $[H]^2$ è costante.

Il Teorema di Ramsey ha anche la seguente versione finita.

Teorema di Ramsey Finito Per ogni $m \in \mathbf{N}$ esiste un $n \in \mathbf{N}$ tale che per ogni colorazione in 2 colori delle coppie di un insieme X di n elementi esiste un sottinsieme $Y \subseteq X$ tale che Y ha m elementi e f è costante su $[Y]^2$ (ossia tutte le coppie in Y hanno lo stesso colore).

Dare una dimostrazione della versione finita usando la versione infinita e il Teorema di Compattezza.

Consideriamo il linguaggio $\{R, B\}$ con R e B due simboli di relazione a 2 posti, il cui significato intuitivo è il seguente: $R(x, y)$ sta per "la coppia $\{x, y\}$ ha il colore 0" e $B(x, y)$ sta per "la coppia $\{x, y\}$ ha il colore 1".

Ragionare per assurdo assumendo che la versione finita del Teorema di Ramsey sia falsa. Dunque esiste un m tale che per ogni n esiste una colorazione f_n in 2 colori di un insieme di n elementi che non ammette alcun sottinsieme di grandezza m sulle cui coppie f_n sia costante.

Fissiamo un tale m.

Scrivere una teoria T che esprima i seguenti fatti:

(1) R e B sono una colorazione delle coppie del dominio.
(2) Per ogni $n \in \mathbf{N}$ un enunciato che esprima: Non esiste un insieme di n elementi contenenti un sottinsiame di m elementi su cui la colorazione è costante.

Argomentare che T è finitamente soddisfacibile (usando l'esistenza degli f_n).

Concludere che T è soddisfacibile.

Argomentare che un modello di T "contiene" una colorazione di un insieme numerabile in 2 colori che contraddice la verità del Teorema Infinito di Ramsey.

Esercizio 6.4. Dimostrare che la proprietà di essere un grafo non-biaprtito non è finitamente assiomatizzabile.

(Suggerimento: usare una caratterizzazione dei grafi bipartiti in termini di cicli e il Teorema di Compattezza).

Esercizio 6.5. Sia $<$ un ordine totale stretto su un insieme X (i.e., $<$ è anti-simmetrico, irriflessivo e totale). Una tale relazione è un *buon ordinamento* se non ammette sequenze infinite decrescenti. Dimostrare per Compattezza che la proprietà di essere un buon ordinamento non è assiomatizzabile.

Esercizio 6.6. Chiamiamo albero un insieme A parzialmente ordinato da una relazione $\leq$, con un elemento distinto r (detto radice) tale che per ogni elemento $a \in A$ l'insieme degli antenati di a, $\{x \in A : x \neq a \text{ e } a \leq x\}$ consiste in una sequenza finita $a_1, \ldots, a_n$ tale che $a < a_1 < a_2 < \cdots < a_n < r$, (dove scriviamo $x < y$ per $x \leq y$ e $x \neq y$).

Dimostrare che la nozione di albero non è assiomatizzabile.

Esercizio 6.7. Siano $p_0, p_1, p_2, \ldots$ i numeri primi in ordine crescente. Dimostrare che per ogni sottinsieme $S \subseteq \mathbf{N}$ esiste un modello dell'aritmetica (i.e. un modello di tutti gli enunciati veri nel modello standard) che contiene un elemento a tale che a è divisibile per p_i per tutti e soli i p_i con $i \in S$.

(Suggerimento: una nuova costante e il Teorema di Compattezza).

Esercizio 6.8. Sia T una teoria che ha modelli finiti e infiniti. Sia E un enunciato tale che: se $\mathfrak{A} \models T$ e $\mathfrak{A}$ è infinito allora $\mathfrak{A} \models E$. Dimostrare che esiste un $b \in \mathbf{N}$ tale che: Se $\mathfrak{A} \models T$ e $\mathfrak{A}$ ha cardinalità $\geq b$ allora $\mathfrak{A} \models E$.

(Suggerimento: Compattezza e suoi corollari).

Esercizio 6.9. Esiste una teoria T con modelli infiniti, almeno un modello finito ma che non possiede modelli finiti di grandezza arbitraria?

Esercizio 6.10. Sia $\mathfrak{A}$ un modello non-standard dell'aritmetica e sia $F(x)$ una formula con una variabile libera. Se per infiniti $n \in \mathbf{N}$ abbiamo $\mathfrak{A} \models F[\binom{x}{\bar{n}^{\mathfrak{A}}}]$ allora esiste un numero non-standard $a \in A$ tale che $\mathfrak{A} \models F[\binom{x}{a}]$. Ossia: se una formula è soddisfatta da infiniti numeri standard allora è soddisfatta da un non-standard.

(Suggerimento: ragionare sulle proprietà degli insiemi definibili nel modello non-standard).

7. Incompletezza

8. Incompletezza

Esercizio 8.1. Dimostrare che la ω-coerenza di una teoria implica la coerenza.

Esercizio 8.2. Sia G l'enunciato di Gödel per una teoria formale $T \supseteq \mathsf{MA}$. Dimostrare che se T è coerente allora $T + \{\neg G\}$ è coerente ma non ω-coerente.

Esercizio 8.3. Sia $\mathcal{L}$ il linguaggio di **MA**. Sia c una nuova costante. Consideriamo la teoria T ottenuta aggiungendo a **MA** tutti gli enunciati $\neg(c = \bar{n})$, per ogni $n \in \mathbf{N}$. Assumendo che **MA** sia coerente, possiamo dire se T è coerente? T è ω-coerente?

Esercizio 8.4. Se T è una teoria incoerente (nel linguaggio dell'aritmetica) allora ogni insieme $X \subseteq \mathbf{N}$ è esprimibile in T. (Si ricorda che un insieme X è esprimibile in T se esiste una formula $F(x)$ con una sola variabile libera tale che per ogni $n \in \mathbf{N}$: se $n \in X$ allora $T \vdash F(\bar{n})$, e se $n \notin X$ allora $T \vdash \neg F(\bar{n})$).

Esercizio 8.5. Sia T una teoria decidibile (ossia: esiste un algoritmo per decidere se una formula è in T o no). Se un insieme $S \subseteq \mathbf{N}$ è esprimibile in T allora S è computabilmente enumerabile (ossia esiste un algoritmo per produrre una lista di tutti e soli gli elementi di S).

Esercizio 8.6. Sia T una teoria nel linguaggio dell'aritmetica. Sia $S \subseteq \mathbf{N}$ un insieme non calcolabile (i.e., la funzione caratteristica di S non è calcolabile). Supponiamo che esista una formula $F(x)$ con una unica variabile libera nel linguaggio di T tale che F esprime S in T. Allora T è coerente. (Suggerimento: si tenga conto del fatto che un insieme X è calcolabile se e solo se X e $\mathbf{N} \setminus X$ sono entrambi computabilmente enumerabili).

Esercizio 8.7. Consideriamo il seguente enunciato dovuto a Rosser (cfr. dispense):

$$E := (\forall y)(F(\overline{m}, y) \to (\exists z)(z \leq y \wedge H(\overline{m}, z))),$$

dove m è il codice della formula $(\forall y)(F(x, y) \to (\exists z)(z \leq y \wedge H(x, z)))$ e F rappresenta la relazione R delle dispense e H rappresenta la relazione S delle dispense (cfr. dispensa 9).

Dimostrare che se $T \supseteq \mathsf{MA}$ è una teoria formale coerente allora $T \nvdash E$ e $T \nvdash \neg E$.

Esercizio 8.8. Una teoria T è detta ω-incompleta se esiste una formula $F(x)$ con una sola variabile libera tale che per tutti gli $n \in \mathbf{N}$ si ha $T \vdash F(\bar{n})$ ma $T \nvdash \forall x F(x)$. Dimostrare che se T è coerente, allora è ω-incompleta.

Esercizio 8.9. (*) Un quantificatore limitato è un quantificatore che compare in un contesto del tipo

$$(\forall x)(x \leq t \to A)$$

oppure

$$(\exists x)(x \leq t \wedge A)$$

dove t è un termine non contenente x e A è una formula qualunque.

Una formula è detta di tipo Σ_1 se è di forma $\exists x_1 \ldots \exists x_n A$ con A contenente soltanto quantificatori limitati (o nessuna quantificazione).

Una teoria T è detta 1-coerente se vale la seguente proprietà:

Per ogni formula $R(x)$ senza quantificatori illimitati, se per ogni $n \in \mathbf{N}$ vale $T \vdash R(\overline{n})$, allora $T \nvdash \exists x \neg R(x)$.

Sia T una teoria nel linguaggio dell'aritmetica con la seguente proprietà:

Per ogni enunciato E di tipo Σ_1, se $\mathbf{N} \models E$ allora $T \vdash E$.

Dimostrare che, per ogni enunciato A, $T \cup \{A\}$ è 1-coerente se e solo se per ogni enunciato E di tipo Π_1 vero in $\mathbf{N}$, $T \cup \{A, E\}$ è coerente.

(Un enunciato è di tipo Π_1 se è di forma $\forall x_1 \ldots \forall x_k H$ dove H contiene solo quantificatori illimitati. Si noti che la negazione di una formula Σ_1 è di tipo Π_1, e viceversa).

On formally undecidable propositions of *Principia Mathematica* and related systems I

Kurt Gödel

1931

0 About this document

Gödel's famous proof [2, 1] is highly interesting, but may be hard to understand. Some of this difficulty is due to the fact that the notation used by Gödel has been largely replaced by other notation. Some of this difficulty is due to the fact that while Gödel's formulations are concise, they sometimes require the readers to make up their own interpretations for formulae, or to keep definitions in mind that may not seem mnemonic to them.

This document is a translation of a large part of Gödel's proof. The translation happens on three levels:

- from German to English
- from Gödel's notation to more common mathematical symbols
- from paper to hyper-text

Hyper-text and colors are used as follows: definitions take place in blue italics, like this: *defined term*. Wherever the defined term is used, we have a red hyper-link to the place in the text where the term was first defined, like this: defined term. Furthermore, each defined term appears in the clickable index at the end of this document. In the margin of the document, there are page-numbers like this [173], which refer to the original document. Here are links for looking up something by page: 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 196. Finally, small text snippets in magenta are comments not present in the original text, but perhaps useful for the reader.

This translation omits all foot-notes from the original, and only contains sections 1 and 2 (out of four).

The translation comes as-is, with no explicit or implied warranty. Use at your own risk, the translator is not willing to take any responsibility for problems you might have because of errors in the translation, or because of misunderstandings. You are permitted to reproduce this document all you like, but only if you include this notice.

Boulder, November 27, 2000 *Martin Hirzel*

[173]

On formally undecidable propositions of *Principia Mathematica* and related systems I

Kurt Gödel

1931

1 Introduction

The development of mathematics towards greater exactness has, as is well-known, lead to formalization of large areas of it such that you can carry out proofs by following a few mechanical rules. The most comprehensive current formal systems are the system of *Principia Mathematica (PM)* on the one hand, the Zermelo-Fraenkelian axiom-system of set theory on the other hand. These two systems are so far developed that you can formalize in them all proof methods that are currently in use in mathematics, i.e. you can reduce these proof methods to a few axioms and deduction rules. Therefore, the conclusion seems plausible that these deduction rules are sufficient to decide *all* mathematical questions expressible in those systems. We will show that this is not true, but that there are even relatively easy problem in the theory of ordinary whole numbers that can not be decided from the axioms. This is not due to the nature of these systems, [174] but it is true for a very wide class of formal systems, which in particular includes all those that you get by adding a finite number of axioms to the above mentioned systems, provided the additional axioms don't make false theorems provable.

Let us first sketch the main intuition for the proof, without going into detail and of course without claiming to be exact. The formulae of a formal system (we will restrict ourselves to the *PM* here) can be viewed syntactically as finite sequences of the basic symbols (variables, logical constants, and parentheses or separators), and it is easy to define precisely *which* sequences of the basic symbols are syntactically correct formulae and which are not. Similarly, *proofs* are formally nothing else than finite sequences of formulae (with specific definable properties). Of course, it is irrelevant for meta-mathematical observations what signs are taken for basic symbols, and so we will chose natural numbers for them. Hence, a formula is a finite sequence of natural numbers, and a proof schema is a finite sequence of finite sequences of natural numbers. The meta-mathematical concepts (theorems) hereby become concepts (theorems) about natural numbers, which makes them (at least partially) expressible in the symbols of the system *PM*. In particular, one can show that the concepts "formula", "proof schema", "provable formula" are all expressible within the system *PM*, i.e. one can, for example,

come up with a formula $F(v)$ of *PM* that has one free variable v (whose type is sequence of numbers) such that the semantic interpretation of $F(v)$ is: v is a provable formula. We will now construct an undecidable theorem of the system *PM*, i.e. a theorem A for which neither A nor $\neg A$ is provable, as follows:

We will call a formula of *PM* with exactly one free variable of type natural numbers a *class-sign*. We will assume the class-signs are somehow numbered, call the nth one R_n, and note that both the concept "class-sign" and the ordering relation R are definable within the system *PM*. Let α be an arbitrary class-sign; with $\alpha(n)$ we denote the formula that you get when you substitute n for the free variable of α. Also, the ternary relation $x \Leftrightarrow y(z)$ is definable within *PM*. Now we will define a class K of natural numbers as follows: [175]

$$K = \{n \in \mathbb{N} \mid \neg \mathit{provable}(R_n(n))\} \tag{1}$$

(where $\mathit{provable}(x)$ means x is a provable formula). With other words, K is the set of numbers n where the formula $R_n(n)$ that you get when you insert n into its own formula R_n is improvable. Since all the concepts used for this definition are themselves definable in *PM*, so is the compound concept K, i.e. there is a class-sign S such that the formula $S(n)$ states that $n \in K$. As a class-sign, S is identical with a specific R_q, i.e. we have

$$S \Leftrightarrow R_q$$

for a specific natural number q. We will now prove that the theorem $R_q(q)$ is undecidable within *PM*. We can understand this by simply plugging in the definitions: $R_q(q) \Leftrightarrow S(q) \Leftrightarrow q \in K \Leftrightarrow \neg \mathit{provable}(R_q(q))$, in other words, $R_q(q)$ states "I am improvable." Assuming the theorem $R_q(q)$ were provable, then it would also be true, i.e. because of (1) $\neg \mathit{provable}(R_q(q))$ would be true in contradiction to the assumption. If on the other hand $\neg R_q(q)$ were provable, then we would have $q \notin K$, i.e. $\mathit{provable}(R_q(q))$. That means that both $R_q(q)$ and $\neg R_q(q)$ would be provable, which again is impossible.

The analogy of this conclusion with the Richard-antinomy leaps to the eye; there is also a close kinship with the liar-antinomy, because our undecidable theorem $R_q(q)$ states that q is in K, i.e. according to (1) that $R_q(q)$ is not provable. Hence, we have in front of us a theorem that states its own unprovability. The proof method we just applied is obviously applicable to any formal system that on the one hand is expressive enough to allow the definition of the concepts used above (in particular the concept "provable formula"), and in which on the other hand all provable formulae are also true. The following exact implementation of the proof will among other things have the goal to replace the second prerequisite by a purely formal and much weaker one. [176]

From the remark that $R_q(q)$ states its own improvability it immediately follows that $R_q(q)$ is correct, since $R_q(q)$ *is* in fact unprovable (because it is undecidable). The theorem which is undecidable *within the system PM* has hence been decided by meta-mathematical considerations. The exact analysis of this strange fact leads to surprising results about consistency proofs for formal systems, which will be discussed in section 4 (theorem XI).

2 Main Result

We will now exactly implement the proof sketched above, and will first give an exact description of the formal system P, for which we want to show the existence of undecidable theorems. By and large, P is the system that you get by building the logic of *PM* on top the Peano axioms (numbers as individuals, successor-relation as undefined basic concept).

2.1 Definitions

The *basic signs* of system P are the following:

I. Constant: "$\neg$" (not), "$\vee$" (or), "$\forall$" (for all), "0" (zero), "*succ*" (the successor of), "(", ")" (parentheses). Gödel's original text uses a different notation, but the reader may be more familiar with the notation adapted in this translation.

II. *Variable of type one* (for individuals, i.e. natural numbers including 0): "x_1", "y_1", "z_1", ...
Variables of type two (for classes of individuals, i.e. subsets of $\mathbb{N}$): "x_2", "y_2", "z_2", ...
Variables of type three (for classes of classes of individuals, i.e. sets of subsets of $\mathbb{N}$): "x_3", "y_3", "z_3", ...
And so on for every natural number as type.

Remark: Variables for binary or n-ary functions (relations) are superfluous as basic signs, because one can define relations as classes of ordered pairs and ordered pairs as classes of classes, e.g. the ordered pair (a, b) by $\{\{a\}, \{a, b\}\}$, where $\{x, y\}$ and $\{x\}$ stand for the classes whose only elements are x, y and x, respectively.

By a *sign of type one* we understand a combination of signs of the form [177]

$$a,\ succ(a),\ succ(succ(a)),\ succ(succ(succ(a))),\ \ldots \quad \text{etc.},$$

where a is either 0 or a variable of type one. In the first case we call such a sign a *number-sign*. For $n > 1$ we will understand by a *sign of type n* a variable of type n. We call combinations of signs of the form $a(b)$, where b is a sign of type n and a a sign of type $n + 1$, *elementary formulae*. We define the class of *formulae* as the smallest set that contains all elementary formulae and that contains for a, b always also $\neg(a)$, $(a) \vee (b)$, $\forall x\,.\,(a)$ (where x is an arbitrary variable). We call $(a) \vee (b)$ the *disjunction* of a and b, $\neg(a)$ the *negation* and $\forall x\,.\,(a)$ the *generalization* of a. A formula that contains no free variables (where *free variables* is interpreted in the usual manner) is called *proposition-formula*. We call a formula with exactly n free individual-variables (and no other free variables) an *n-ary relation sign*, for $n = 1$ also class-sign.

By **subst**$a\binom{v}{b}$ (where a is a formula, v is a variable, and b is a sign of the same type as v) we understand the formula that you get by substituting b for every free occurrence

of v in a. We say that a formula a is a *type-lift* of another formula b if you can obtain a from b by increasing the type of all variables occurring in a by the same number.

The following formulae (I through V) are called *axioms* (they are written with the help of the abbreviations (defined in the usual manner) $\wedge$, $\Rightarrow$, $\Leftrightarrow$, $\exists x$, $=$, and using the customary conventions for leaving out parentheses):

I. The *Peano axioms*, which give fundamental properties for natural numbers.

1. $\neg(succ(x_1) = 0)$ We start to count at 0.
2. $succ(x_1) = succ(y_1) \Rightarrow x_1 = y_1$ If two natural numbers $x_{1,2} \in \mathbb{N}$ have the same successor, they are equal.
3. $\Big(x_2(0) \wedge \forall x_1 \,.\, x_2(x_1) \Rightarrow x_2(succ(x_1))\Big) \Rightarrow \forall x_1 \,.\, x_2(x_1)$ We can prove a predicate x_2 on natural numbers by natural induction.

II. Every formula obtained by inserting arbitrary formulae for p, q, r in the following schemata. We call these *proposition axioms*. [178]

1. $p \vee p \Rightarrow p$
2. $p \Rightarrow p \vee q$
3. $p \vee q \Rightarrow q \vee p$
4. $(p \Rightarrow q) \Rightarrow (r \vee p \Rightarrow r \vee q)$

III. Every formula obtained from the two schemata

1. $(\forall v \,.\, a) \Rightarrow \mathbf{subst} a \binom{v}{c}$
2. $(\forall v \,.\, b \vee a) \Rightarrow (b \vee \forall v \,.\, a)$

by inserting the following things for a, v, b, c (and executing the operation denoted by **subst** in 1.):

Insert an arbitrary formula for a, an arbitrary variable for v, any formula where v does not occur free for b, and for c a sign of the same type as v with the additional requirement that c does not contain a free variable that would be bound in a position in a where v is free.

For lack of a better name, we will call these *quantor axioms*.

IV. Every formula obtained from the schema

1. $\exists u \,.\, \forall v \,.\, (u(v) \Leftrightarrow a)$

by inserting for v and u any variables of type n and $n+1$ respectively and for a a formula that has no free occurrence of u. This axiom takes the place of the *reducibility axiom* (the *comprehension axiom* of set theory).

V. Any formula obtained from the following by type-lift (and the formula itself):

1. $\Big(\forall x_1 \,.\, (x_2(x_1) \Leftrightarrow y_2(x_1))\Big) \Rightarrow x_2 = y_2$

   This axiom states that a class is completely determined by its elements. Let us call it the *set axiom*.

A formula c is called the *immediate consequence* of a and b (of a) if a is the formula $\neg b \vee c$ (or if c is the formula $\forall v \,.\, a$, where v is any variable). The class of *provable formulae* is defined as the smallest class of formulae that contains the axioms and is closed under the relation "immediate consequence".

2.2 Gödel-numbers

We will now uniquely associate the primitive signs of system P with natural numbers as follows: [179]

$$\begin{array}{lll} \text{“0”} \ldots 1 & \text{“}succ\text{”} \ldots 3 & \text{“}\neg\text{”} \ldots 5 \\ \text{“}\vee\text{”} \ldots 7 & \text{“}\forall\text{”} \ldots 9 & \text{“(”} \ldots 11 \\ & & \text{“)”} \ldots 13 \end{array}$$

Furthermore we will uniquely associate each variable of type n with a number of the form p^n (where p is a prime > 13). Thus there is a one-to-one-correspondence between every finite string of basic signs and a sequence of natural numbers. We now map the sequences of natural numbers (again in one-to-one correspondence) to natural numbers by having the sequence $n_1, n_2, \ldots, n_k$ correspond to the number $2^{n_1} \cdot 3^{n_2} \cdot \ldots \cdot p_k^{n_k}$, where p_k is the kth prime (by magnitude). Thus, there is not only a uniquely associated natural number for every basic sign, but also for every sequence of basic signs. We will denote the number associated with the basic sign (resp. the sequence of basic signs) a by $\Phi(a)$. Now let $R(a_1, a_2, \ldots, a_n)$ be a given class or relation between basic signs or sequences of them. We will associate that with the class (relation) $R'(x_1, x_2, \ldots, x_n)$ that holds between $x_1, x_2, \ldots, x_n$ if and only if there are $a_1, a_2, \ldots, a_n$ such that for $i = 1, 2, \ldots, n$ we have $x_i = \Phi(a_i)$ and the $R(a_1, a_2, \ldots, a_n)$ holds. We will denote the classes and relations on natural numbers which are associated with the meta-mathematical concepts, e.g. "variable", "formula", "proposition-formula", "axiom", "provable formula" etc., in the above mentioned manner, by the same word in small caps. The proposition that there are undecidable problems in system P for example reads like this: There are PROPOSITION-FORMULAE a, such that neither a nor the NEGATION of a is a PROVABLE FORMULA.

2.3 Primitive recursion

At this point, we will make an excursion to make an observation that *a priori* does not have anything to do with the system P, and will first give the following definition: we

say a number-theoretical formula $\phi(x_1, x_2, \ldots, x_n)$ is *defined via primitive recursion* in terms of the number-theoretical formulae $\psi(x_1, x_2, \ldots, x_{n-1})$ and $\mu(x_1, x_2, \ldots, x_{n+1})$ if the following holds for all $x_2, \ldots, x_n, k$:

$$\begin{aligned} \phi(0, x_2, \ldots, x_n) &= \psi(x_2, \ldots, x_n), \\ \phi(k+1, x_2, \ldots, x_n) &= \mu(k, \phi(k, x_2, \ldots, x_n), x_2, \ldots, x_n) \end{aligned} \tag{2}$$

We call a number-theoretical formula ϕ *primitive recursive* if there is a finite sequence of number-theoretical formulae $\phi_1, \phi_2, \ldots, \phi_n$ ending in ϕ such that every function ϕ_k of the sequence is either defined from two of the preceding formulae by primitive recursion or results by inserting into any of the preceding ones or, and this is the base case, is a constant or the successor function $succ(x) = x + 1$. The length of the shortest sequence of ϕ_i belonging to a primitive recursive function ϕ is called its *degree*. We call a relation $R(x_1, \ldots, x_n)$ *primitive recursive* if there is a primitive recursive function $\phi(x_1, \ldots, x_n)$ such that for all $x_1, x_2, \ldots, x_n$, [180]

$$R(x_1, \ldots, x_n) \Leftrightarrow (\phi(x_1, \ldots, x_n) = 0).$$

The following theorems hold:

I. *Every function (relation) that you get by inserting primitive recursive functions in the places of variables of other primitive recursive functions (relations) is itself primitive recursive; likewise every function that you get from primitive recursive functions by the schema (2).*

II. *If R and S are primitive recursive relations, then so are $\neg R, R \vee S$ (and therefore also $R \wedge S$).*

III. *If the functions $\phi(\vec{x}), \psi(\vec{y})$ are primitive recursive, then so is the relation $\phi(\vec{x}) = \psi(\vec{y})$. We have resorted to a vector notation $\vec{x}$ to denote finite-length tuples of variables.*

IV. *If the function $\phi(\vec{x})$ and the relation $R(y, \vec{z})$ are primitive recursive, then so are the relations S, T*

$$S(\vec{x}, \vec{z}) \Leftrightarrow \left(\exists y \leq \phi(\vec{x}) \,.\, R(y, \vec{z})\right)$$

$$T(\vec{x}, \vec{z}) \Leftrightarrow \left(\forall y \leq \phi(\vec{x}) \,.\, R(y, \vec{z})\right)$$

as well as the function ψ

$$\psi(\vec{x}, \vec{z}) = \left(\mathbf{argmin}\, y \leq \phi(\vec{x}) \,.\, R(y, \vec{z})\right)$$

where **argmin** $x \leq f(x) \,.\, F(x)$ *stands for the smallest x for which $(x \leq f(x)) \wedge F(x)$ holds, and for 0 if there is no such number. Readers to whom an operational*

description appeals more may want to think of this as a loop that tries every value from 1 to $\phi(\vec{x})$ to determine the result. The crucial point here is this theorem does not state that an unbounded loop (or recursion) is primitive recursive; those are in fact strictly more powerful in terms of computability.

Theorem I follows immediately from the definition of "primitive recursive". Theorems II and III are based upon the fact that the number-theoretical functions

$$\alpha(x), \beta(x,y), \gamma(x,y)$$

corresponding to the logical concepts $\neg, \vee, =$ (where $n = 0$ is taken for **true** and $n \neq 0$ for **false**), namely

$$\alpha(x) = \begin{cases} 1 & \text{for } x = 0 \\ 0 & \text{for } x \neq 0 \end{cases}$$

$$\beta(x,y) = \begin{cases} 0 & \text{if one or both of } x, y \text{ are } = 0 \\ 1 & \text{if both } x, y \text{ are } \neq 0 \end{cases}$$

$$\gamma(x,y) = \begin{cases} 0 & \text{if } x = y \\ 1 & \text{if } x \neq y \end{cases}$$

are primitive recursive, as one can easily convince oneself. The proof for theorem IV is, in short, the following: by assumption there is a primitive recursive $\rho(y, \vec{z})$ such that: [181]

$$R(y, \vec{z}) \Leftrightarrow (\rho(y, \vec{z}) = 0)$$

Using the recursion-schema (2) we now define a function $\chi(y, \vec{z})$ as follows:

$$\chi(0, \vec{z}) = 0$$
$$\chi(n+1, \vec{z}) = (n+1) \cdot A + \chi(n, \vec{z}) \cdot \alpha(A)$$

where $A = \alpha\Big(\alpha(\rho(0, \vec{z}))\Big) \cdot \alpha\Big(\rho(n+1, \vec{z})\Big) \cdot \alpha\Big(\chi(n, \vec{z})\Big)$.

A, which makes use of the above defined α and of the fact that a product is 0 if one of its factors is 0, can be described by the following pseudo-code:

$A =$ **if**$(\rho(0, \vec{z}) = 0)$
then 0
else if$(\rho(n+1, \vec{z}) \neq 0)$
then 0
else if$(\chi(n, \vec{z}) \neq 0)$
then 0
else 1

It is a nice example for how arithmetic can be used to emulate logics.

Therefore, $\chi(n+1, \vec{z})$ is either $= n+1$ (if $A = 1$) or $= \chi(n, \vec{z})$ (if $A = 0$). Obviously, the first case will occur if and only if all factors of A are 1, i.e. if we have

$$\neg R(0, \vec{z}) \wedge R(n+1, \vec{z}) \wedge (\chi(n, \vec{z}) = 0).$$

This implies that the function $\chi(n, \vec{z})$ (viewed as a function of n) remains 0 up to the smallest value of n for which $R(n, \vec{z})$ holds, and has that value from then on (if $R(0, \vec{z})$ already holds then $\chi(n, \vec{z})$ is correspondingly constant and $= 0$). Therefore, we have

$$\psi(\vec{x}, \vec{z}) = \chi(\phi(\vec{x}), \vec{z})$$
$$S(\vec{x}, \vec{z}) \Leftrightarrow R(\psi(\vec{x}, \vec{z}), \vec{z}).$$

It is easy to reduce the relation T to a case analogous to that of S by negation. This concludes the proof of theorem IV.

2.4 Expressing metamathematical concepts

As one can easily convince oneself, the functions $x + y$, $x \cdot y$, x^y and furthermore the relations $x < y$ and $x = y$ are primitive recursive. For example, the function $x + y$ can be constructed as $0 + y = y$ and $(k + 1) + y = \mathit{succ}(k + y)$, i.e. $\psi(y) = y$ and $\mu(k, l, y) = \mathit{succ}(l)$ in schema (2). Using these concepts, we will now define a sequence of functions (relations) 1-45, each of which is defined from the preceding ones by the methods given by theorems I through IV. In doing so, usually multiple of the definition steps allowed by theorems I through IV are combined in one. Each of the functions (relations) 1-45, among which we find for example the concepts "FORMULA", "AXIOM", and "IMMEDIATE CONSEQUENCE", is therefore primitive recursive. [182]

1. $y \mid x \Leftrightarrow \exists z \leq x \,.\, x = y \cdot z$
x is divisible by y.

2. $\mathit{isPrime}(x) \Leftrightarrow \neg\Big(\exists z \leq x \,.\, (z \neq 1 \wedge z \neq x \wedge z \mid x)\Big) \wedge \Big(x > 1\Big)$
x is a prime number.

3. $\mathit{prFactor}(0, x) = 0$
$\mathit{prFactor}(n+1, x) = \mathbf{argmin}\, y \leq x \,.\, \Big(\mathit{isPrime}(y) \wedge y \mid x \wedge y > \mathit{prFactor}(n, x)\Big)$
$\mathit{prFactor}(n, x)$ is the nth (by size) prime number contained in x.

4. $0! = 1$
$(n+1)! = (n+1) \cdot n!$

5. $\mathit{nthPrime}(0) = 0$
$\mathit{nthPrime}(n{+}1) = \mathbf{argmin}\, y \leq (\mathit{nthPrime}(n)!{+}1) \,.\, \Big(\mathit{isPrime}(y) \wedge y > \mathit{nthPrime}(n)\Big)$
$\mathit{nthPrime}(n)$ is the nth (by size) prime number.

6. $item(n,x) = \mathbf{argmin}\, y \leq x \,.\, \Big((prFactor(n,x)^y \mid x) \wedge \neg(prFactor(n,x)^{y+1} \mid x)\Big)$

$item(n,x)$ is the nth item of the sequence of numbers associated with x (for $n > 0$ and n not larger than the length of this sequence).

7. $length(x) = \mathbf{argmin}\, y \leq x \,.\, \Big(prFactor(y,x) > 0 \wedge prFactor(y+1,x) = 0\Big)$

$length(x)$ is the length of the sequence of numbers associated with x.

8. $x \circ y = \mathbf{argmin}\, z \leq nthPrime(length(x) + length(y))^{x+y} \,.$
$(\forall n \leq length(x) \,.\, item(n,z) = item(n,x)) \wedge$
$(\forall 0 < n \leq length(y) \,.\, item(n + length(x), z) = item(n,y))$

$x \circ y$ corresponds to the operation of "concatenating" two finite sequences of numbers.

9. $seq(x) = 2^x$

$seq(x)$ corresponds to the number sequence that consists only of the number x (for $x > 0$).

10. $paren(x) = seq(11) \circ x \circ seq(13)$

$paren(x)$ corresponds to the operation of "parenthesizing" (11 and 13 are associated with the primitive signs "(" and ")").

11. $vtype(n,x) \Leftrightarrow \Big(\exists 13 < z \leq x \,.\, isPrime(z) \wedge x = z^n\Big) \wedge n \neq 0$

x is a VARIABLE OF TYPE n.

12. $isVar(x) \Leftrightarrow \exists n \leq x \,.\, vtype(n,x)$

x is a VARIABLE.

13. $not(x) = seq(5) \circ paren(x)$

$not(x)$ is the NEGATION of x.

[183]

14. $or(x,y) = paren(x) \circ seq(7) \circ paren(y)$

$or(x,y)$ is the DISJUNCTION of x and y.

15. $forall(x,y) = seq(9) \circ seq(x) \circ paren(y)$

$forall(x,y)$ is the GENERALIZATION of y by the VARIABLE x (provided that x is a VARIABLE).

16. $succ\_n(0,x) = x$
$succ\_n(n+1,x) = seq(3) \circ succ\_n(n,x)$

$succ\_n(n,x)$ corresponds to the operation of "prepending the sign '$succ$' in front of x for n times".

17. $number(n) = succ(n, seq(1))$

$number(n)$ is the NUMBER-SIGN for the number n.

18. $stype_1(x) \Leftrightarrow \exists m,n \leq x \,.$
$(m = 1 \vee vtype(1,m)) \wedge x = succ\_n(n, seq(m))$

x is a SIGN OF TYPE ONE.

19. $stype(n,x) \Leftrightarrow \big(n = 1 \wedge stype_1(x)\big) \vee \big(n > 1 \wedge \exists v \leq x\,.\,(vtype(n,v) \wedge x = R(v))\big)$

x is a SIGN OF TYPE n.

20. $elFm(x) \Leftrightarrow \exists y, z, n \leq x\,.\, (stype(n,y) \wedge stype(n+1,z) \wedge x = z \circ paren(y))$

x is an ELEMENTARY FORMULA.

21. $op(x,y,z) \Leftrightarrow (x = not(y)) \vee (x = or(y,z)) \vee (\exists v \leq x\,.\, isVar(v) \wedge x = forall(v,y))$

22. $fmSeq(x) \Leftrightarrow \big(\forall 0 < n \leq length(x)\,.\; elFm(item(n,x)) \vee \exists 0 < p, q < n\,.\, op(item(n,x), item(p,x), item(q,x))\big) \wedge length(x) > 0$

x is a sequence of FORMULAE, each of which is either an ELEMENTARY FORMULA or is obtained from the preceding ones by the operations of NEGATION, DISJUNCTION, or GENERALIZATION.

23. $isFm(x) \Leftrightarrow \exists n \leq \big(nthPrime(length(x)^2)\big)^{x\cdot(length(x))^2}\,.\; fmSeq(n) \wedge x = item(length(n), n)$

x is a FORMULA (i.e. the last item of a sequence n of formulae).

24. $bound(v,n,x) \Leftrightarrow isVar(v) \wedge isFm(x) \wedge \exists a, b, c \leq x\,.\; x = a \circ forall(v,b) \circ c \wedge isFm(b) \wedge length(a) + 1 \leq n \leq length(a) + length(forall(v,b))$

The variable v is BOUND in x at position n.

[184]

25. $free(v,n,x) \Leftrightarrow isVar(v) \wedge isFm(x) \wedge v = item(n,x) \wedge n \leq length(x) \wedge \neg bound(v,n,x)$

The variable v is FREE in x at position n.

26. $free(v,x) \Leftrightarrow \exists n \leq length(x)\,.\, free(v,n,x)$

v occurs in x as a FREE VARIABLE.

27. $insert(x,n,y) = \mathbf{argmin}\, z \leq (nthPrime(length(x) + length(y)))^{x+y}\,.\; \exists u, v \leq x\,.\; x = u \circ seq(item(n,x)) \circ v \wedge z = u \circ y \circ v \wedge n = length(u) + 1$

You obtain $insert(x,n,y)$ from x by inserting y instead of the nth item in the sequence x (provided that $0 < n \leq length(x)$).

28. $freePlace(0,v,x) = \mathbf{argmin}\, n \leq length(x)\,.\; free(v,n,x) \wedge \neg\exists n < p \leq length(x)\,.\, free(v,p,x)$
$freePlace(k+1,v,x) = \mathbf{argmin}\, n < freePlace(n,k,v)\,.\; free(v,n,x) \wedge \neg\exists n < p < freePlace(n,k,v)\,.\, free(v,p,x)$

$freePlace(k,v,x)$ is the $k+1$st place in x (counted from the end of FORMULA x) where v is FREE (and 0 if there is no such place).

29. $nFreePlaces(v,x) = \mathbf{argmin}\, n \leq length(x)\,.\, freePlace(n,v,x) = 0$
$nFreePlaces(v,x)$ is the number of places where v is free in x.

30. $subst'(0,x,v,y) = x$
$subst'(k+1,x,v,y) = insert(subst'(k,x,v,y), freePlace(k,v,x), y)$

31. $subst(x,v,y) = subst'(nFreePlaces(v,x),x,v,y)$
$subst(x,v,y)$ is the above defined concept $\mathbf{subst}a\binom{v}{b}$.

32. $imp(x,y) = or(not(x),y)$
$and(x,y) = not(or(not(x),not(y)))$
$equiv(x,y) = and(imp(x,y),imp(y,x))$
$exists(v,y) = not(forall(v,not(y)))$

33. $typeLift(n,x) = \mathbf{argmin}\, y \leq x^{x^n}\,.$
$\forall k \leq length(x)\,.$
$item(k,x) \leq 13 \wedge item(k,y) = item(k,x) \vee$
$item(k,x) > 13 \wedge item(k,y) = item(k,x) \cdot prFactor(1,item(k,x))^n$
$typeLift(n,x)$ is the nth TYPE-LIFT of x (if x and $typeLift(n,x)$ are FORMULAE).

There are three specific numbers corresponding to the axioms I, 1 to 3 (the Peano axioms), which we will denote by pa_1, pa_2, pa_3, and we define:

34. $peanoAxiom(x) \Leftrightarrow (x = pa_1 \vee x = pa_2 \vee x = pa_3)$ [185]

35. $prop1Axiom(x) \Leftrightarrow \exists y \leq x\,.\, isFm(y) \wedge x = imp(or(y,y),y)$
x is a FORMULA that has been obtained by inserting into the axiom schema II, 1. We define $prop2Axiom(x)$, $prop3Axiom(x)$, and $prop4Axiom(x)$ analogously.

36. $propAxiom(x) \Leftrightarrow prop1Axiom(x) \vee prop2Axiom(x) \vee prop3Axiom(x) \vee prop4Axiom(x)$
x is a FORMULA that has been obtained by inserting into on of the proposition axioms.

37. $quantor1AxiomCondition(z,y,v) \Leftrightarrow \neg\exists n \leq length(y), m \leq length(z), w \leq z\,.$
$w = item(m,z) \wedge bound(w,n,y) \wedge free(v,n,y)$
z does not contain a VARIABLE that is BOUND anywhere in y where v is FREE. This condition for the applicability of axiom III, 1, ensured that a substitution of z for the free occurrences of v in y does not accidentally bind some of z's variables.

38. $quantor1Axiom(x) \Leftrightarrow \exists v,y,z,n \leq x\,.$
$vtype(n,v) \wedge stype(n,z) \wedge isFm(y) \wedge quantor1AxiomCondition(z,y,v) \wedge$
$x = imp(forall(v,y), subst(y,v,z))$
x is a FORMULA obtained by substitution from the axiom schema III, 1, i.e. one of the quantor axioms.

39. $quantor2Axiom(x) \Leftrightarrow \exists v,q,p \leq x\,.$
$isVar(v) \wedge isFm(p) \wedge \neg free(v,p) \wedge isFm(q) \wedge$

$x = imp(forall(v, or(p, q)), or(p, forall(v, q)))$

x is a FORMULA obtained by substitution from the axiom schema III, 2, i.e. the other one of the quantor axioms.

40. $reduAxiom(x) \Leftrightarrow \exists u, v, y, n \leq x \,.$
$vtype(n, v) \wedge vtype(n+1, u) \wedge \neg free(u, y) \wedge isFm(y) \wedge$
$x = exists(u, forall(v, equiv(seq(u) \circ paren(seq(v)), y)))$

x is a FORMULA obtained by substitution from the axiom schema IV, 1, i.e. from the reducibility axiom.

There is a specific number corresponding to axiom V, 1, (the set axiom), which we will denote by sa, and we define:

41. $setAxiom(x) \Leftrightarrow \exists n \leq x \,.\, x = typeLift(n, sa)$

42. $isAxiom(x) \Leftrightarrow peanoAxiom(x) \vee propAxiom(x) \vee$
$quantor1Axiom(x) \vee quantor2Axiom(x) \vee reduAxiom(x) \vee$
$setAxiom(x)$

x is an AXIOM.

43. $immConseq(x, y, z) \Leftrightarrow y = imp(z, x) \vee \exists v \leq x \,.\, isVar(v) \wedge x = forall(v, y)$

x is an IMMEDIATE CONSEQUENCE of y and z.

[186]

44. $isProofFigure(x) \Leftrightarrow \big(\forall 0 < n \leq length(x) \,.$
$isAxiom(item(n, x)) \vee \exists 0 < p, q < n \,.$
$immConseq(item(n, x), item(p, x), item(q, x))\big) \wedge$
$length(x) > 0$

x is a PROOF FIGURE (a finite sequence of FORMULAE, each of which is either an AXIOM or the IMMEDIATE CONSEQUENCE of two of the preceding ones).

45. $proofFor(x, y) \Leftrightarrow isProofFigure(x) \wedge item(length(x), x) = y$

x is a PROOF for the FORMULA y.

46. $provable(x) \Leftrightarrow \exists y \,.\, proofFor(y, x)$

x is a PROVABLE FORMULA. ($provable(x)$ is the only one among the concepts 1-46 for which we can not assert that it is primitive recursive).

2.5 Denotability and provability

The fact that can be expressed vaguely by: Every primitive recursive relation is definable within system P (interpreting that system as to content), will be expressed in the following theorem *without* referring to the interpretation of formulae of P:

Theorem V: *For every primitive recursive relation $R(x_1, \ldots, x_n)$ there is a* RELATION SIGN r *(with the* FREE VARIABLES $u_1, \ldots, u_n$*), such that for each n-tuple $(x_1, \ldots, x_n)$ the following holds:*

$$R(x_1, \ldots, x_n) \Rightarrow \mathit{provable}(\mathit{subst}(r, u_1 \ldots u_n, \mathit{number}(x_1) \ldots \mathit{number}(x_n))) \quad (3)$$

$$\neg R(x_1, \ldots, x_n) \Rightarrow \mathit{provable}(\mathit{not}(\mathit{subst}(r, u_1 \ldots u_n, \mathit{number}(x_1) \ldots \mathit{number}(x_n)))) \quad (4)$$

We contend ourselves with giving a sketchy outline of the proof for this theorem here, since it does not offer any difficulties in principle and is rather cumbersome. We prove the theorem for all relations $R(x_1, \ldots, x_n)$ of the form $x_1 = \phi(x_2, \ldots, x_n)$ (where ϕ is a primitive recursive function) and apply natural induction by ϕ's degree. For functions of degree one (i.e. constants and the function $x+1$) the theorem is trivial. Hence, let ϕ be of degree m. It is built from functions of lower degree $\phi_1, \ldots, \phi_k$ by the operations of insertion and primitive recursive definition. Since everything has already been proven for $\phi_1, \ldots, \phi_k$ by the inductive assumption, there are corresponding RELATION SIGNS $r_1, \ldots, r_k$ such that (3), (4) hold. The definition processes by which ϕ is built from $\phi_1, \ldots, \phi_k$ (insertion and primitive recursion) can all be modeled formally in system P. Doing this, one gets from $r_1, \ldots, r_k$ a new RELATION SIGN r for which one can proof the validity of (3), (4) without difficulties. A RELATION SIGN r associated with a primitive recursive relation in this manner shall be called *primitive recursive*. [187]

2.6 Undecidability theorem

We now come to the goal of our elaborations. Let κ be any class of FORMULAE. We denote with $\mathit{Conseq}(\kappa)$ the smallest set of FORMULAE that contains all FORMULAE of κ and all AXIOMS and is closed under the relation "IMMEDIATE CONSEQUENCE". κ is called *ω-consistent* if there is no CLASS-SIGN a such that

$$\Big(\forall n \,.\, \mathit{subst}(a, v, \mathit{number}(n)) \in \mathit{Conseq}(\kappa)\Big) \wedge \mathit{not}(\mathit{forall}(v, a)) \in \mathit{Conseq}(\kappa)$$

where v is the FREE VARIABLE of the CLASS-SIGN a. With other words, a witness against ω-consistency would be a formula a with one free variable where we can derive $a(n)$ for all n, but also $\neg\forall n \,.\, a(n)$, a contradiction.

Every ω-consistent system is, of course, also consistent. The reverse, however, does not hold true, as will be shown later. We call a system *consistent* if there is no formula a such that both a and $\neg a$ are provable. Such a formula would be a witness against the consistency, but in general not against the ω-consistency. With other words, ω-consistency is stronger than consistency: the first implies the latter, but not vice versa.

The general result about the existence of undecidable propositions goes as follows:

Theorem VI: *For every ω-consistent primitive recursive class κ of* FORMULAE *there is a primitive recursive* CLASS-SIGN *r such that neither $\mathit{forall}(v, r)$ nor $\mathit{not}(\mathit{forall}(v, r))$ belongs to $\mathit{Conseq}(\kappa)$ (where v is the* FREE VARIABLE *of r).*

Since the premise in the theorem is ω-consistency, which is stronger than consistency, the theorem is less general than if its premise were just consistency.

Proof: Let κ be any ω-consistent primitive recursive class of FORMULAE. We define:

$$\begin{array}{l} \mathit{isProofFigure}_\kappa(x) \Leftrightarrow \\ \quad \Big(\forall n \leq \mathit{length}(x) \,.\, \mathit{isAxiom}(\mathit{item}(n,x)) \vee (\mathit{item}(n,x) \in \kappa) \vee \\ \qquad \exists 0 < p,q < n \,.\, \mathit{immedConseq}(\mathit{item}(n,x), \mathit{item}(p,x), \mathit{item}(q,x))\Big) \wedge \\ \quad \mathit{length}(x) > 0 \end{array} \tag{5}$$

(compare to the analogous concept 44)

$$\mathit{proofFor}_\kappa(x,y) \Leftrightarrow \mathit{isProofFigure}_\kappa(x) \wedge \mathit{item}(\mathit{length}(x),x) = y \tag{6}$$

$$\mathit{provable}_\kappa(x) \Leftrightarrow \exists y \,.\, \mathit{proofFor}_\kappa(y,x) \tag{6.1}$$

(compare to the analogous concepts 45, 46).

The following obviously holds:

$$\forall x \,.\, \Big(\mathit{provable}_\kappa(x) \Leftrightarrow x \in \mathit{Conseq}(\kappa)\Big), \tag{7}$$

$$\forall x \,.\, \Big(\mathit{provable}(x) \Rightarrow \mathit{provable}_\kappa(x)\Big). \tag{8}$$

Now we define the relation: [188]

$$Q(x,y) \Leftrightarrow \neg\Big(\mathit{proofFor}_\kappa(x, \mathit{subst}(y,19,\mathit{number}(y)))\Big). \tag{8.1}$$

Intuitively $Q(x,y)$ means x does not prove $y(y)$.

Since $\mathit{proofFor}_\kappa(x,y)$ (by (6), (5)) and $\mathit{subst}(y,19,\mathit{number}(y))$ (by definitions 17, 31) are primitive recursive, so is $Q(x,y)$. According to theorem V we hence have a RELATION SIGN q (with the FREE VARIABLES 17, 19) such that the following holds:

$$\begin{array}{c} \neg \mathit{proofFor}_\kappa(x, \mathit{subst}(y,19,\mathit{number}(y))) \Rightarrow \\ \mathit{provable}_\kappa(\mathit{subst}(q,17\ 19,\mathit{number}(x)\ \mathit{number}(y))) \end{array} \tag{9}$$

$$\begin{array}{c} \mathit{proofFor}_\kappa(x, \mathit{subst}(y,19,\mathit{number}(y))) \Rightarrow \\ \mathit{provable}_\kappa(\mathit{not}(\mathit{subst}(q,17\ 19,\mathit{number}(x)\ \mathit{number}(y)))). \end{array} \tag{10}$$

We set:

$$p = \mathit{forall}(17,q) \tag{11}$$

(p is a CLASS-SIGN with the FREE VARIABLE 19 (which intuitively means 19(19), i.e. $y(y)$, is improvable)) and

$$r = \mathit{subst}(q,19,\mathit{number}(p)) \tag{12}$$

(r is a primitive recursive CLASS-SIGN with the FREE VARIABLE 17 (which intuitively means that 17, i.e. x, does not prove $p(p)$, where $p(p)$ means $p(p)$ is unprovable)).

Then the following holds:

$$\begin{aligned} subst(p, 19, number(p)) &= subst(forall(17, q), 19, number(p)) \\ &= forall(17, subst(q, 19, number(p))) \\ &= forall(17, r) \end{aligned} \tag{13}$$

(because of (11 and 12)); furthermore:

$$subst(q, 17\ 19, number(x)\ number(p)) = subst(r, 17, number(x)) \tag{14}$$

(because of (14)). The recurring $forall(17, r)$ can be interpreted as there is no prove for $p(p)$, with other words, $forall(17, r)$ states that the statement $p(p)$ that states its own improvability is improvable. If we now insert p for y in (9) and (10), we get, taking (13) and (14) into account:

$$\neg proofFor_\kappa(x, forall(17, r)) \Rightarrow provable_\kappa(subst(r, 17, number(x))) \tag{15}$$

$$proofFor_\kappa(x, forall(17, r)) \Rightarrow provable_\kappa(not(subst(r, 17, number(x)))) \tag{16}$$

This yields: [189]

1. $forall(17, r)$ is not κ-PROVABLE. Because if that were the case, there would (by (7)) exist an n such that $proofFor_\kappa(n, forall(17, r))$. By (16) we would hence have:

   $$provable_\kappa(not(subst(r, 17, number(n)))),$$

   while on the other hand the κ-PROVABILITY of $forall(17, r)$ also implies that of $subst(r, 17, number(n))$. Therefore κ would be inconsistent (and in particular ω-inconsistent).

2. $not(forall(17, r))$ is not κ-PROVABLE. Proof: As has just been shown, $forall(17, r)$ is not κ-PROVABLE, i.e. (by (7)) we have

   $$\forall n\,.\,\neg proofFor_\kappa(n, forall(17, r)).$$

   This implies by (15)

   $$\forall n\,.\,provable_\kappa(subst(r, 17, number(n)))$$

   which would, together with

   $$provable_\kappa(not(forall(17, r))),$$

   contradict the ω-consistency of κ.

Therefore $forall(17, r)$ is not decidable from κ, whereby theorem VI is proved.

2.7 Discussion

One can easily convince oneself that the proof we just did is constructive, i.e. it the following is intuitionistically flawlessly proven:
Let any primitive recursively defined class κ of formulae be given. Then if the formal decision (from κ) of the PROPOSITION-FORMULA $forall(17, r)$ is also given, one can effectively present:

1. A PROOF for $not(forall(17, r))$.

2. For any given n a PROOF for $subst(r, 17, number(n))$, i.e. a formal decision for $forall(17, r)$ would imply the effective presentability of an ω-inconsistency-proof.

Let us call a relation (class) between natural numbers $R(x_1, \ldots, x_n)$ *decision-definite* if there is an n-ary RELATION SIGN r such that (3) and (4) (c.f. theorem V) hold. In particular therefore every primitive recursive relation is by Theorem V decision-definite. Analogously, a RELATION SIGN shall be called *decision-definite* if it corresponds to a decision-definite relation in this manner. For the existence of propositions undecidable from κ it is now sufficient to require of a class κ that it is ω-consistent and decision-definite. With other words, it is not even important how the class of added axioms κ is defined, we just have to be able to decide with the means of the system whether something is an axiom or not. This is because the decision-definiteness carries over from κ to $poofFor_\kappa(x, y)$ (compare to (5), (6)) and to $Q(x, y)$ (compare to (9)), and only that was used for the above proof. In this case, the undecidable theorem takes on the form $forall(v, r)$, where r is a decision-definite CLASS-SIGN (by the way, it is even sufficient that κ is decision-definite in the system augmented by κ). [190]

If instead of ω-consistency we only assume consistency for κ, then, although the existence of an undecidable proposition does not follow, there follows the existence of a property (r) for which a counter-example is not *presentable* and neither is it provable that the relation holds for all numbers. Because for the proof that $forall(17, r)$ is not ω-PROVABLE we only used the ω-consistency of κ (compare to page 189), and $\neg provable_\kappa(forall(17, r))$ implies by (15) for each number x that $subst(r, 17, number(x))$ holds, i.e. that for no number $not(subst(r, 17, number(x)))$ is provable.

If you add $not(forall(17, r))$ to κ you get a consistent but not ω-consistent class of FORMULAE κ'. κ' is consistent because otherwise $forall(17, r)$ would be provable. But κ' is not ω-consistent, since because of $\neg provable_\kappa(forall(17, r))$ and (15) we have

$$\forall x \,.\, provable_\kappa(subst(r, 17, number(x))),$$

and hence in particular

$$\forall x \,.\, provable_{\kappa'}(subst(r, 17, number(x))),$$

and on the other hand of course

$$provable_{\kappa'}(\neg forall(17, r)).$$

But that means that $forall(17, r)$ precisely fits the definition of a witness against ω-consistency.

A special case of theorem VI is the theorem where the class κ consists of a finite number of FORMULAE (and perhaps the ones derived from these by TYPE-LIFT). Every finite class κ is of course primitive recursive. Let a be the largest contained number. Then we have for κ in this case

$$x \in \kappa \Leftrightarrow \exists m \leq x, n \leq a \,.\, n \in \kappa \wedge x = typeLift(m, n)$$

Hence, κ is primitive recursive. This allows us to conclude for example that also with the help of the axiom of choice (for all types) or the generalized continuum hypothesis not all propositions are decidable, assuming that these hypotheses are ω-consistent.

During the proof of theorem VI we did not use any other properties of the system P than the following:

1. The class of axioms and deduction rules (i.e. the relation "immediate consequence") are primitive recursively definable (as soon as you replace the basic signs by numbers in some way).

2. Every primitive recursive relation is definable within the system P (in the sense of theorem V).

Hence there are undecidable propositions of the form $\forall x \,.\, F(x)$ in every formal system that fulfills the preconditions 1, 2 and is ω-consistent, and also in every extension of such a system by a primitive recursively definable, ω-consistent class of axioms. To this kind of systems belong, as one can easily confirm, the Zermelo-Fraenkelian axiom-system and the von Neumannian system of set-theory, furthermore the axiom-system of number-theory which consists of the Peano axioms, primitive recursive definition (by schema (2)) and the logical deduction rules. Simply every system whose deduction rules are the usual ones and whose axioms (analogously like in P) are made by insertion into a finite number of schemas fulfills precondition 1. [191]

3 Generalizations

—omitted— [196]

4 Implications for the nature of consistency

—omitted—

A Experiences

This translation was done for a reason. I took Mike Eisenberg's class "Computer Science: The Canon" at the University of Colorado in fall 2000. It was announced as a "great works" lecture-and-discussion course, offering an opportunity to be pointed to some great papers, giving an incentive to read them, and providing a forum for discussion. This also explains my motivation for translating Gödel's proof: it is a truly impressive and fascinating paper, there is some incentive in completing my final paper for a class, and this exercise should and did benefit me intellectually. Here, I will try to share the experiences I made doing the translation. I deliberately chose a personal, informal style for this final section to stress that what I write here are just my opinions, nothing less and nothing more.

A.1 Have I learned or gained something?

First of all, how useful is it to read this paper anyway, whether you translate it or not? One first answer that comes to mind is that the effort of understanding it hones abstract thinking skills, and that some basic concepts like Peano axioms, primitive recursion, or consistency are nicely illustrated and shown in a motivated context. The difficulty with this argument is that it is self-referential: we read this paper to hone skills that we would not need to hone if we would not read this kind of papers. I don't really have a problem with that, people do many things for their own sake, but fortunately there are other gains to be had from reading this paper. One thing that I found striking is how I only fully appreciated the thoughts from section 2.7 on re-reading the proof. I find it fascinating just how general the result is: your formal system does not need to be finite, or even primitive recursively describable, no, it suffices that you can decide its set of axioms in itself. I am not sure whether other people are as fascinated by this as I am; if you are not, try to see what I mean, it's worth the effort! But in any case, I have gained an appreciation for the beauty and power of the results, which I am happy for. Finally, there is some hope that the writing skills, proof techniques, and thought processes exhibited by this paper might rub off, so to speak. Part of learning an art is to study the masters, and Gödel was clearly a master in his art!

Second, how useful was the translation itself? Well, it was useful to try out some ideas I had about how translating a technical paper between languages might work. My recipe was to first read and understand the whole paper, then translate it one sentence at a time (avoid to start translating a sentence before having a plan for all of it!), and finally to read it fast to check the flow and logic. This might or might not be the best way, but it worked well enough for me. For understanding the paper itself, the translation between languages or the use of hyper-text as a medium did not help me much as I was doing it. More important was the translation of notation to one I am more used to, and the occasional comment to express my view of a tricky detail. Last but not least, it seems hardly necessary to admit my strong affection for type-setting, and there is a certain pleasure in looking over and polishing something you crafted that

I believe I share with many people.

A.2 Has the paper improved?

The original paper is brilliant, well-written, rich of content, relevant. Yet I went ahead and tinkered around, changing a little thing here and there, taking much more freedom than the translators for [2, 1]. Yes, the paper did improve! It is closer to my very personal ideas of what it should ideally look like. To me who did the changes just a few days ago the modified paper looks better than the original.

In section 0, I announced a translation along three dimensions, namely language (from German to English), notation (using symbols I am more used to) and medium (exploiting hyper-text). Let us review each one in turn and criticize the changes.

Language. After finishing the translation, I compared it with the ones in [2, 1], and found that although they are different, the wording probably does not matter all that much. To give an example where it did seem to play a role, here are three wordings for *besteht eine nahe Verwandtschaft*: (i) *is closely related*, (ii) *is also a close relationship*, (iii) *is also a close kinship*. The third one is mine, and my motivation for it is that it is the most punchy one, for what it's worth. The stumbling blocks in this kind of translation, as I see it, are rather the technical terms that may be in no dictionary. For example, I could well imagine that my translation *decision-definite* seems unnatural to someone studying logic who might be used to another term.

Notation. This is the part of the translation that I believe helps the most in making the paper more accessible for readers with a similar educational background as mine. For example, I have never seen "Π" used for "∀", and inside an English text I find "$bound(v, n, x)$" easier to parse than "$v\,\mathrm{Geb}\,n, x$".

Hyper-text. During the translation, it became clearer to me just how very "hyper-text" the paper already was! On the one hand, the fact that one naturally refers back to definitions and theorems underlines that hyper-text is a natural way of presentation. On the other hand, the fact that one gets along quite well with a linear text, relying on the readers to construct the thought-building in their own head, seems to suggest that the change of medium was in fact rather superficial. I would be interested in the opinions of readers of this document on this: did the hyper-text improve the paper?

Clearly, the most important aspects of the paper are still the organization, writing, and explanation skills of Gödel himself. And clearly, the paper is still an intellectual challenge, yielding its rewards only to the fearless. To assume that my work has changed either of these facts significantly would be presumptuous.

A.3 Opinions

This discussion is a trade-off between being careful and thoughtful on the one hand, and being forthcoming and fruitful on the other. As it leans more to the second half of the spectrum, I might as well go ahead and state some opinions the project and the reflections upon it have inspired in me.

- A well-written technical paper already has the positive features of hyper-text. This may not seem so at first glance, but compare it to the typical web-page and then ask yourself which has more coherence. To me, coherence is part of the essence and beauty of cross-referencing.
- There is an analogy between writing papers and computer programs, and it is amazing how far you can stretch it without it breaks down. The skill of gradually building up your vocabulary, dividing and conquering the task in a clean and skillful way, and commenting on what you do are all illustrated nicely by Gödel's proof.
- Reading and understanding Gödel's proof yields many benefits. There are pearls to be found in its contents, and skills to be practiced that go beyond what one might think at first glance.

I am well aware that I did not give many arguments to support these opinions. That would be the stuff for a paper by itself, and the reader is encouraged to think about them. But above all, enjoy the paper "On formally undecidable propositions of Principia Mathematica and related systems I" itself, which after all makes up the main part of this document!

Literatur

[1] Kurt Gödel. *On Formally Undecidable Propositions of Principia Mathematica and Related Systems*. Dover, 1962.

[2] Kurt Gödel. Über formal unentscheidbare Sätze der *Principia Mathematica* und verwandter Systeme I. In Solomon Feferman, editor, *Kurt Gödel: Collected Works*, volume 1, pages 144–195. Oxford University Press, 1986. German text, parallel English translation.

B Index

THE JOURNAL OF SYMBOLIC LOGIC
Volume 14, Number 3, Sept. 1949

THE COMPLETENESS OF THE FIRST-ORDER FUNCTIONAL CALCULUS

LEON HENKIN[1]

Although several proofs have been published showing the completeness of the propositional calculus (cf. Quine (1)[2]), for the first-order functional calculus only the original completeness proof of Gödel (2) and a variant due to Hilbert and Bernays have appeared. Aside from novelty and the fact that it requires less formal development of the system from the axioms, the new method of proof which is the subject of this paper possesses two advantages. In the first place an important property of formal systems which is associated with completeness can now be generalized to systems containing a non-denumerable infinity of primitive symbols. While this is not of especial interest when formal systems are considered as *logics*—i.e., as means for analyzing the structure of languages—it leads to interesting applications in the field of abstract algebra. In the second place the proof suggests a new approach to the problem of completeness for functional calculi of higher order. Both of these matters will be taken up in future papers.

The system with which we shall deal here will contain as primitive symbols

$$(\quad)\quad \supset\quad f\quad ,$$

and certain sets of symbols as follows:

(i) *propositional symbols* (some of which may be classed as *variables*, others as *constants*), and among which the symbol "f" above is to be included as a constant;

(ii) for each number $n = 1, 2, \cdots$ a set of *functional symbols of degree n* (which again may be separated into *variables* and *constants*); and

(iii) *individual symbols* among which *variables* must be distinguished from *constants*. The set of variables must be infinite.

Elementary well-formed formulas are the propositional symbols and all formulas of the form $G(x_1, \cdots, x_n)$ where G is a functional symbol of degree n and each x_i is an individual symbol.

Well-formed formulas (wffs) consist of the elementary well-formed formulas together with all formulas built up from them by repeated application of the following methods:

(i) If A and B are wffs so is $(A \supset B)$;

(ii) If A is a wff and x an individual variable then $(x)A$ is a wff. Method (ii) for forming wffs is called *quantification with respect to the variable x*. Any occurrence of the variable x in the formula $(x)A$ is called *bound*. Any occurrence of a symbol which is not a bound occurrence of an individual variable according to this rule is called *free*.

Received August 6, 1948.

[1] This paper contains results of research undertaken while the author was a National Research Council predoctoral fellow. The material is included in "The Completeness of Formal Systems," a thesis presented to the faculty of Princeton University in candidacy for the degree of Doctor of Philosophy.

[2] Numbers refer to items in the bibliography appearing at the end of the paper.

In addition to formal manipulation of the formulas of this system we shall be concerned with their *meaning* according to the following interpretation. The propositional constants are to denote one of the truth values, T or F, the symbol "f" denoting F, and the propositional variables are to have the set of these truth values as their range. Let an arbitrary set, I, be specified as a domain of individuals, and let each individual constant denote a particular element of this domain while the individual variables have I as their range. The functional constants (variables) of degree n are to denote (range over) subsets of the set of all ordered n-tuples of I. $G(x_1, \cdots, x_n)$ is to have the value T or F according as the n-tuple $\langle x_1, \cdots, x_n \rangle$ of individuals is or is not in the set G; $(A \supset B)$ is to have the value F if A is T and B is F, otherwise T; and $(x)A$ is to have the value T just in case A has the value T for every element x in I.[3]

If A is a wff, I a domain, and if there is some assignment of denotations to the constants of A and of values of the appropriate kind to the variables with free occurrences in A, such that for this assignment A takes on the value T according to the above interpretation, we say that A is *satisfiable with respect to* I. If *every* such assignment yields the value T for A we say that A is *valid with respect to* I. A is *valid* if it is valid with respect to every domain. We shall give a set of axioms and formal rules of inference adequate to permit formal proofs of every valid formula.

Before giving the axioms, however, we describe certain rules of abbreviation which we use to simplify the appearance of wffs and formula schemata. If A is any wff and x any individual variable we write

$$\sim A \quad \text{for} \quad (A \supset f),$$

$$(\exists x)A \quad \text{for} \quad \sim(x)\sim A.$$

From the rules of interpretation it is seen that $\sim A$ has the value T or F according as A has the value F or T, while $(\exists x)A$ denotes T just in case there is some individual x in I for which A has the value T.

Furthermore we may omit outermost parentheses, replace a left parenthesis by a dot omitting its mate at the same time if its mate comes at the end of the formula (except possibly for other right parentheses), and put a sequence of wffs separated by occurrences of "$\supset$" when association to the left is intended. For example,

$$A \supset B \supset . C \supset D \supset E \quad \text{for} \quad ((A \supset B) \supset ((C \supset D) \supset E)),$$

where A, B, C, D, E may be wffs or abbreviations of wffs.

If A, B, C are any wffs, the following are called *axioms*:

1. $C \supset . B \supset C$
2. $A \supset B \supset . A \supset (B \supset C) \supset . A \supset C$
3. $A \supset f \supset f \supset A$
4. $(x)(A \supset B) \supset . A \supset (x)B$, where x is any individual variable with no free occurrence in A.

[3] A more precise, syntactical account of these ideas can be formulated along the lines of Tarski (3). But this semantical version will suffice for our purposes.

5. $(x)A \supset B$, where x is any individual variable, y any individual symbol, and B is obtained by substituting y for each free occurrence of x in A, provided that no free occurrence of x in A is in a well-formed part of A of the form $(y)C$.

There are two formal rules of inference:

I (*Modus Ponens*). To infer B from any pair of formulas A, $A \supset B$.

II (*Generalization*). To infer $(x)A$ from A, where x is any individual variable.

A finite sequence of wffs is called a *formal proof from assumptions* Γ, where Γ is a set of wffs, if every formula of the sequence is either an axiom, an element of Γ, or else arises from one or two previous formulas of the sequence by *modus ponens* or generalization, except that no variable with a free occurrence in some formula of Γ may be generalized upon. If A is the last formula of such a sequence we write $\Gamma \vdash A$. Instead of $\{\Gamma, A\} \vdash B$ ($\{\Gamma, A\}$ denoting the set formed from Γ by adjoining the wff A), we shall write $\Gamma, A \vdash B$. If Γ is the empty set we call the sequence simply a *formal proof* and write $\vdash A$. In this case A is called a *formal theorem*. Our object is to show that every valid formula is a formal theorem, and hence that our system of axioms and rules is *complete*.

The following theorems about the first-order functional calculus are all either well-known and contained in standard works, or else very simply derivable from such results. We shall use them without proof here, referring the reader to Church (4) for a fuller account.

III (*The Deduction Theorem*). If $\Gamma, A \vdash B$ then $\Gamma \vdash A \supset B$ (for any wffs A, B and any set Γ of wffs).

6. $\vdash B \supset f \supset . B \supset C$
7. $\vdash B \supset . C \supset f \supset . B \supset C \supset f$
8. $\vdash (x)(A \supset f) \supset . (\exists x)A \supset f$
9. $\vdash (x)B \supset f \supset . (\exists x)(B \supset f)$.

IV. If Γ is a set of wffs no one of which contains a free occurrence of the individual symbol u, if A is a wff and B is obtained from it by replacing each free occurrence of u by the individual symbol x (none of these occurrences of x being bound in B), then if $\Gamma \vdash A$, also $\Gamma \vdash B$.

This completes our description of the formal system; or, more accurately, of a class of formal systems, a certain degree of arbitrariness having been left with respect to the nature and number of primitive symbols.

Let $\boldsymbol{S}_0$ be a particular system determined by some definite choice of primitive symbols. A set Λ of wffs of $\boldsymbol{S}_0$ will be called *inconsistent* if $\Lambda \vdash f$, otherwise *consistent*. A set Λ of wffs of $\boldsymbol{S}_0$ will be said to be *simultaneously satisfiable* in some domain I of individuals if there is some assignment of denotations (values) of the appropriate type to the constants (variables) with free occurrences in formulas of Λ, for which each of these formulas has the value T under the interpretation previously described.

THEOREM. *If Λ is a set of formulas of $\mathbf{S}_0$ in which no member has any occurrence of a free individual variable, and if Λ is consistent, then Λ is simultaneously satisfiable in a domain of individuals having the same cardinal number as the set of primitive symbols of $\mathbf{S}_0$.*

We shall carry out the proof for the case where $\mathbf{S}_0$ has only a denumerable infinity of symbols, and indicate afterward the simple modifications needed in the general case.

Let u_{ij} $(i, j = 1, 2, 3, \cdots)$ be symbols not occurring among the symbols of $\mathbf{S}_0$. For each i $(i = 1, 2, 3, \cdots)$ let $\mathbf{S}_i$ be the first-order functional calculus whose primitive symbols are obtained from those of $\mathbf{S}_{i-1}$ by adding the symbols u_{ij} $(j = 1, 2, 3, \cdots)$ as individual constants. Let $\mathbf{S}_\omega$ be the system whose symbols are those appearing in any one of the systems $\mathbf{S}_i$. It is easy to see that the wffs of S_ω are denumerable, and we shall suppose that some particular enumeration is fixed on so that we may speak of the first, second, $\cdots$, nth, $\cdots$ formula of $\mathbf{S}_\omega$ in the standard ordering.

We can use this ordering to construct in $\mathbf{S}_0$ a maximal consistent set of cwffs, Γ_0, which contains the given set Λ. (We use "cwff" to mean *closed* wff: *a* wff *which contains no free occurrence of any individual variable.*) Γ_0 is maximal consistent in the sense that if A is any cwff of $\mathbf{S}_0$ which is not in Γ_0, then $\Gamma_0, A \vdash f$; but not $\Gamma_0 \vdash f$.

To construct Γ_0 let Γ_{00} be Λ and let B_1 be the first (in the standard ordering) cwff A of $\mathbf{S}_0$ such that $\{\Gamma_{00}, A\}$ is consistent. Form Γ_{01} by adding B_1 to Γ_{00}. Continue this process as follows. Assuming that Γ_{0i} and B_i have been found, let B_{i+1} be the first cwff A (of $\mathbf{S}_0$) after B_i, such that $\{\Gamma_{0i}, A\}$ is consistent; then form Γ_{0i+1} by adding B_{i+1} to Γ_{0i}. Finally let Γ_0 be composed of those formulas appearing in any Γ_{0i} $(i = 0, 1, \cdots)$. Clearly Γ_0 *contains* Λ. Γ_0 is *consistent*, for if $\Gamma_0 \vdash f$ then the formal proof of f from assumptions Γ_0 would be a formal proof of f from some finite subset of Γ_0 as assumptions, and hence for some $i(i = 0, 1, \cdots)$ $\Gamma_{0i} \vdash f$ contrary to construction of the sets of Γ_{0i}. Finally, Γ_0 is *maximal* consistent because if A is a cwff of $\mathbf{S}_0$ such that $\{\Gamma_0, A\}$ is consistent then surely $\{\Gamma_{0i}, A\}$ is consistent for each i; hence A will appear in some Γ_{0i} and so in Γ_0.

Having obtained Γ_0 we proceed to the system $\mathbf{S}_1$ and form a set Γ_1 of its cwffs as follows. Select the first (in the standard ordering) cwff of Γ_0 which has the form $(\exists x)A$ (unabbreviated: $((x)(A \supset f) \supset f)$), and let A' be the result of substituting the symbol u_{11} of $\mathbf{S}_1$ for all free occurrences of the variable x in the wff A. The set $\{\Gamma_0, A'\}$ must be a consistent set of cwffs of $\mathbf{S}_1$. For suppose that $\Gamma_0, A' \vdash f$. Then by III (the Deduction Theorem), $\Gamma_0 \vdash A' \supset f$; hence by IV, $\Gamma_0 \vdash A \supset f$; by II, $\Gamma_0 \vdash (x)(A \supset f)$; and so by 8 and I, $\Gamma_0 \vdash (\exists x)A \supset f$. But by assumption $\Gamma_0 \vdash (\exists x)A$. Hence modus ponens gives $\Gamma_0 \vdash f$ contrary to the construction of Γ_0 as a consistent set.

We proceed in turn to each cwff of Γ_0 having the form $(\exists x)A$, and for the j^{th} of these we add to Γ_0 the cwff A' of $\mathbf{S}_1$ obtained by substituting the constant u_{1j} for each free occurrence of the variable x in the wff A. Each of these adjunctions leaves us with a consistent set of cwffs of $\mathbf{S}_1$ by the argument above.

Finally, after all such formulas A' have been added, we enlarge the resulting set of formulas to a maximal consistent set of cwffs of $\boldsymbol{S}_1$ in the same way that Γ_0 was obtained from Λ in $\boldsymbol{S}_0$. This set of cwffs we call Γ_1.

After the set Γ_i has been formed in the system $\boldsymbol{S}_i$ we construct Γ_{i+1} in $\boldsymbol{S}_{i+1}$ by the same method used in getting Γ_i from Γ_0 but using the constants u_{i+1j} $(j = 1, 2, 3, \cdots)$ in place of u_{1j}. Finally we let Γ_ω be the set of cwffs of $\boldsymbol{S}_\omega$ consisting of all those formulas which are in any Γ_1. It is easy to see that Γ_ω possesses the following properties:

i) Γ_ω is a maximal consistent set of cwffs of $\boldsymbol{S}_\omega$.

ii) If a formula of the form $(\exists x)A$ is in Γ_ω then Γ_ω also contains a formula A' obtained from the wff A by substituting some constant u_{ij} for each free occurrence of the variable x.

Our entire construction has been for the purpose of obtaining a set of formulas with these two properties; they are the only properties we shall use now in showing that the elements of Γ_ω are simultaneously satisfiable in a denumerable domain of individuals.

In fact we take as our domain I simply the set of individual constants of $\boldsymbol{S}_\omega$, and we assign to each such constant (considered as a symbol in an interpreted system) itself (considered as an individual) as denotation. It remains to assign values in the form of truth-values to propositional symbols, and sets of ordered n-tuples of individuals to functional symbols of degree n, in such a way as to lead to a value T for each cwff of Γ_ω.

Every propositional symbol, A, of $\boldsymbol{S}_0$ is a cwff of $\boldsymbol{S}_\omega$; we assign to it the value T or F according as $\Gamma_\omega \vdash A$ or not. Let G be any functional symbol of degree n. We assign to it the class of those n-tuples $\langle a_1, \cdots, a_n\rangle$ of individual constants such that $\Gamma_\omega \vdash G(a_1, \cdots, a_n)$.

This assignment determines a unique truth-value for each cwff of $\boldsymbol{S}_\omega$ under the fundamental interpretation prescribed for quantification and "$\supset$". (We may note that the symbol "f" is assigned F in agreement with that interpretation since Γ_ω is consistent.) We now go on to show the

LEMMA: *For each* cwff A *of* $\boldsymbol{S}_\omega$ *the associated value is* T *or* F *according as* $\Gamma_\omega \vdash A$ *or not.*

The proof is by induction on the length of A. We may notice, first, that if we do not have $\Gamma_\omega \vdash A$ for some cwff A of $\boldsymbol{S}_\omega$ then we do have $\Gamma_\omega \vdash A \supset f$. For by property i) of Γ_ω we would have $\Gamma_\omega, A \vdash f$ and so $\Gamma_\omega \vdash A \supset f$ by III.

In case A is an elementary cwff the lemma is clearly true from the nature of the assignment.

Suppose A is $B \supset C$. If C has the value T, by induction hypothesis $\Gamma_\omega \vdash C$; then $\Gamma_\omega \vdash B \supset C$ by 1 and I. This agrees with the lemma since $B \supset C$ has the value T in this case. Similarly, if B has the value F we do not have $\Gamma_\omega \vdash B$ by induction hypothesis. Hence $\Gamma_\omega \vdash B \supset f$, and $\Gamma_\omega \vdash B \supset C$ by 6 and I. Again we have agreement with the lemma since $B \supset C$ has the value T in this case also. Finally if B and C have the values T and F respectively, so that (induction hypothesis) $\Gamma_\omega \vdash B$ while $\Gamma_\omega \vdash C \supset f$, we must show that $\Gamma_\omega \vdash B$

$\supset C$ does not hold (since $B \supset C$ has the value F in this case). But by 7 and two applications of I we conclude that $\Gamma_\omega \vdash B \supset C \supset f$. Now we see that if $\Gamma_\omega \vdash B \supset C$ then $\Gamma_\omega \vdash f$ by I, contrary to the fact that Γ_ω is consistent (property i).

Suppose A is $(x)B$. If $\Gamma_\omega \vdash (x)B$ then (by 5 and I) $\Gamma_\omega \vdash B'$, where B' is obtained by replacing all free occurrences of x in B by some (arbitrary) individual constant. That is, (induction hypothesis) B has the value T for every individual x of I; therefore A has the value T and the lemma is established in this case. If, on the other hand, we do not have $\Gamma_\omega \vdash (x)B$, then $\Gamma_\omega \vdash (x)B \supset f$ whence (by 9, I) $\Gamma_\omega \vdash (\exists x)(B \supset f)$. Hence, by property ii of Γ_ω, for some individual constant u_{ij} we have $\Gamma_\omega \vdash B' \supset f$, where B' is obtained from B by replacing each free occurrence of x by u_{ij}. Hence for this u_{ij} we cannot have $\Gamma_\omega \vdash B'$ else $\Gamma_\omega \vdash f$ by I contrary to the fact that Γ_ω is consistent (property i). That is, by induction hypothesis, B has the value F for at least the one individual u_{ij} of I and so $(x)B$ has the value F as asserted by the lemma for this case.

This concludes the inductive proof of the lemma. In particular the formulas of Γ_ω all have the value T for our assignment and so are simultaneously satisfiable in the denumerable domain I. Since the formulas of Λ are included among those of Γ_ω our theorem is proved for the case of a system $\boldsymbol{S}_o$ whose primitive symbols are denumerable.

To modify the proof in the case of an arbitrary system $\boldsymbol{S}_o$ it is only necessary to replace the set of symbols u_{ij} by symbols $u_{i\alpha}$, where i ranges over the positive integers as before but α ranges over a set with the same cardinal number as the set of primitive symbols of $\boldsymbol{S}_0$; and to fix on some particular well-ordering of the formulas of the new $\boldsymbol{S}_\omega$ in place of the standard enumeration employed above. (Of course the axiom of choice must be used in this connection.)

The completeness of the system $\boldsymbol{S}_o$ is an immediate consequence of our theorem.

COROLLARY 1: *If A is a valid* wff *of $\boldsymbol{S}_0$ then* $\vdash A$.

First consider the case where A is a cwff. Since A is valid $A \supset f$ has the value F for any assignment with respect to any domain; i.e., $A \supset f$ is not satisfiable. By our theorem it is therefore inconsistent: $A \supset f \vdash f$. Hence $\vdash A \supset f \supset f$ by III and $\vdash A$ by 3 and I.

The case of wff A' which contains some free occurrence of an individual variable may be reduced to the case of the cwff A (the *closure* of A') obtained by prefixing to A' universal quantifiers with respect to each individual variable with free occurrences in A' (in the order in which they appear). For it is clear from the definition of validity that if A' is valid so is A. But then $\vdash A$. From which we may infer $\vdash A'$ by successive applications of 5 and I.

COROLLARY 2: *Let $\boldsymbol{S}_0$ be a functional calculus of first order and* $\mathfrak{m}$ *the cardinal number of the set of its primitive symbols. If Λ is a set of* cwffs *which is simultaneously satisfiable then in particular Λ is simultaneously satisfiable in some domain of cardinal* $\mathfrak{m}$.

This is an immediate consequence of our theorem and the fact that if Λ is simultaneously satisfiable it must also be consistent (since rules of inference

preserve the property of having the value T for any particular assignment in any domain, and so could not lead to the formula f). For the special case where $\mathfrak{m}$ is $\aleph_0$ this corollary is the well-known Skolem-Löwenheim result (5). It should be noticed, for this case, that the assertion of a set of cwffs Λ can no more compel a domain to be finite than non-denumerably infinite: there is always a denumerably infinite domain available. There are also always domains of any cardinality greater than $\aleph_0$ in which a consistent set Λ is simultaneously satisfiable, and sometimes finite domains. However for certain Λ no finite domain will do.

Along with the truth functions of propositional calculus and quantification with respect to individual variables the first-order functional calculus is sometimes formulated so as to include the notion of equality as between individuals. Formally this may be accomplished by singling out some functional constant of degree 2, say Q, abbreviating $Q(x, y)$ as $x = y$ (for individual symbols x, y), and adding the axiom schemata

E1. $x = x$

E2. $x = y \supset . A \supset B$, where B is obtained from A by replacing some free occurrence of x by a free occurrence of y.

For a system S_0' of this kind our theorem holds if we replace "the same cardinal number as" by "a cardinal number not greater than," where the definition of "simultaneously satisfiable" must be supplemented by the provision that the symbol "=" shall denote the relation of equality between individuals. To prove this we notice that a set of cwffs Λ in the system S_0' may be regarded as a set of cwffs (Λ, E_1, E_2) in the system S_0, where E_1 is the set of closures of axioms E_i $(i = 1, 2)$. Since $E_1, E_2 \vdash x = y \supset y = x$ and $E_1, E_2 \vdash x = y \supset . y = z \supset x = z$ we see that the assignment which gives a value T to each formula of Λ, E_1, E_2 must assign some equivalence relation to the functional symbol Q. If we take the domain I' of equivalence classes determined by this relation over the original domain I of constants, and assign to each individual constant (as denotation) the class determined by itself, we are led to a new assignment which is easily seen to satisfy Λ (simultaneously) in S_0'.

A set of wffs may be thought of as a set of axioms determining certain domains as models; namely, domains in which the wffs are simultaneously satisfiable. For a first-order calculus containing the notion of equality we can find axiom sets which restrict models to be finite, unlike the situation for calculi without equality. More specifically, given any finite set of finite numbers there exist axiom sets whose models are precisely those domains in which the number of individuals is one of the elements of the given set. (For example, if the set of numbers is the pair (1, 3) the single axiom

$$(x)(y)(x = y) \vee . (\exists x)(\exists y)(\exists z) . \sim (x = y) \wedge \sim (x = z) \wedge \sim (y = z) \wedge (t) . t = x \vee t = y \vee t = z$$

will suffice, where $A \wedge B$, $A \vee B$ abbreviate $\sim(A \supset \sim B)$, $A \supset B \supset B$ respectively.) However, an axiom set which has models of arbitrarily large finite

cardinality must also possess an infinite model as one sees by considering the formulas

$$C_i: \quad (\exists x_1)(\exists x_2) \cdots (\exists x_i) \,.\, \sim(x_1 = x_2) \wedge \sim(x_1 = x_2) \cdots \wedge \sim(x_{i-1} = x_i).$$

Since by hypothesis any finite number of the C_i are simultaneously satisfiable they are consistent. Hence all the C_i are consistent and so simultaneously satisfiable—which can happen only in an infinite domain of individuals.

There are axiom sets with no finite models—namely, the set of all formulas C_i defined above. Every axiom set with an infinite model has models with arbitrary infinite cardinality. For if α, β range over any set whatever the set of all formulas $\sim(x_\alpha = x_\beta)$ for distinct α, β will be consistent (since the assumption of an infinite model guarantees consistency for any finite set of these formulas) and so can be simultaneously satisfied.

In simplified form the proof of our theorem and corollary 1 may be carried out for the propositional calculus. For this system the symbols u_{ij} and the construction of S_ω may be omitted, an assignment of values being made directly from Γ_0. While such a proof of the completeness of the propositional calculus is short compared with other proofs in the literature the latter are to be preferred since they furnish a constructive method for finding a formal proof of any given tautology, rather than merely demonstrate its existence.

Bibliography

1. Quine, W. V., *Completeness of the propositional calculus*, ***The journal of symbolic logic***, vol. 3 (1938), pp. 37–40.

2. Gödel, Kurt, *Die Vollständigkeit der Axiome des logischen Funktionenkalküls*, ***Monatshefte für Mathematik und Physik***, vol. 37 (1930), pp. 349–360.

3. Tarski, Alfred, *Der Wahrheitsbegriff in den Formalisierten Sprachen*, ***Studia Philosophica***, vol. 1 (1936), pp. 261–405.

4. Church, Alonzo, ***Introduction to Mathematical Logic, Part I***, Annals of Mathematics Studies, Princeton University Press, 1944.

5. Skolem, Th., *Über einige Grundlagenfragen der Mathematik*, ***Skrifter utgitt av Det Norske Videnskaps-Akademi i Oslo***, 1929, no. 4.

PRINCETON UNIVERSITY