Intelligenza Artificiale II

Logica del Primo Ordine

Parte 2: Automazione

Marco Piastra

5

Metodi a tableau

Metodo a tableau, caso proposizionale

- E` un metodo basato sulla refutazione:
  - per stabilire che $\Gamma \models \varphi$ si tenta di mostrare che $\Gamma \cup \{\neg\varphi\}$ è inconsistente

- Caratteristiche generali:
  - l'insieme $\Gamma \cup \{\neg\varphi\}$ viene completamente espanso in forma di *albero*
  - si usano diverse regole di inferenza per l'espansione:
    ogni nodo dell'albero rappresenta una **congiunzione** di fbf
    ogni biforcazione rappresenta una **disgiunzione**
  - un ramo è chiuso non appena si incontra una contraddizione $\{\varphi, \neg\varphi\}$
  - $\Gamma \cup \{\neg\varphi\}$ è inconsistente (o insoddisfacibile)
    se tutti i rami dell'albero sono **chiusi**
  - $\Gamma \cup \{\neg\varphi\}$ <u>non</u> è inconsistente se un ramo non si chiude
  - non è necessario che $\Gamma \cup \{\neg\varphi\}$ sia espresso in una forma normale

Regole per i tableau proposizionali

- Regole α (congiunzione = espansione)

(a1)	(a2)	(a3)	
$\dfrac{\varphi \land \psi}{\varphi, \psi}$	$\dfrac{\neg(\varphi \lor \psi)}{\neg\varphi, \neg\psi}$	$\dfrac{\neg(\neg\varphi)}{\varphi}$	$\dfrac{\neg(\varphi \to \psi)}{\varphi, \neg\psi}$

Per semplicità,
useremo solo le regole
α e β per $\land$, $\lor$, $\neg$

- Regole β (disgiunzione = biforcazione)

(b1)	(b2)			
$\dfrac{\varphi \lor \psi}{\varphi \quad \mid \quad \psi}$	$\dfrac{\neg(\varphi \land \psi)}{\neg\varphi \quad \mid \quad \neg\psi}$	$\dfrac{\varphi \to \psi}{\neg\varphi \quad \mid \quad \psi}$	$\dfrac{\varphi \leftrightarrow \psi}{\neg\varphi, \neg\psi \quad \mid \quad \varphi, \psi}$	$\dfrac{\neg(\varphi \leftrightarrow \psi)}{\neg\varphi, \psi \quad \mid \quad \varphi, \neg\psi}$

Algoritmo per i tableau proposizionali

- Procedura per $\Gamma \models \varphi$, caso proposizionale
  1) Si definisce $\Gamma \cup \{\neg\varphi\}$ come nodo (*tableau*) iniziale
  2) Per ogni nodo, in modalità *depth-first*
     Se il nodo contiene una coppia di letterali complementari, chiudere il ramo
     - se non vi sono altri nodi aperti, la procedura è terminata (successo)
  3) Se il nodo contiene formule composite:
     - applicare le regole α
     - applicare le regole β
  4) Se non si possono applicare regole ed il nodo non è chiuso,
     la procedura è terminata (fallimento)

  – Notare che le regole α vengono applicate prima delle regole β
  – La scelta del metodo di espansione (*depth-first* o *breadth-first*) è irrilevante (*nel caso proposizionale* ..)

Esempio 15

Dimostrare che:

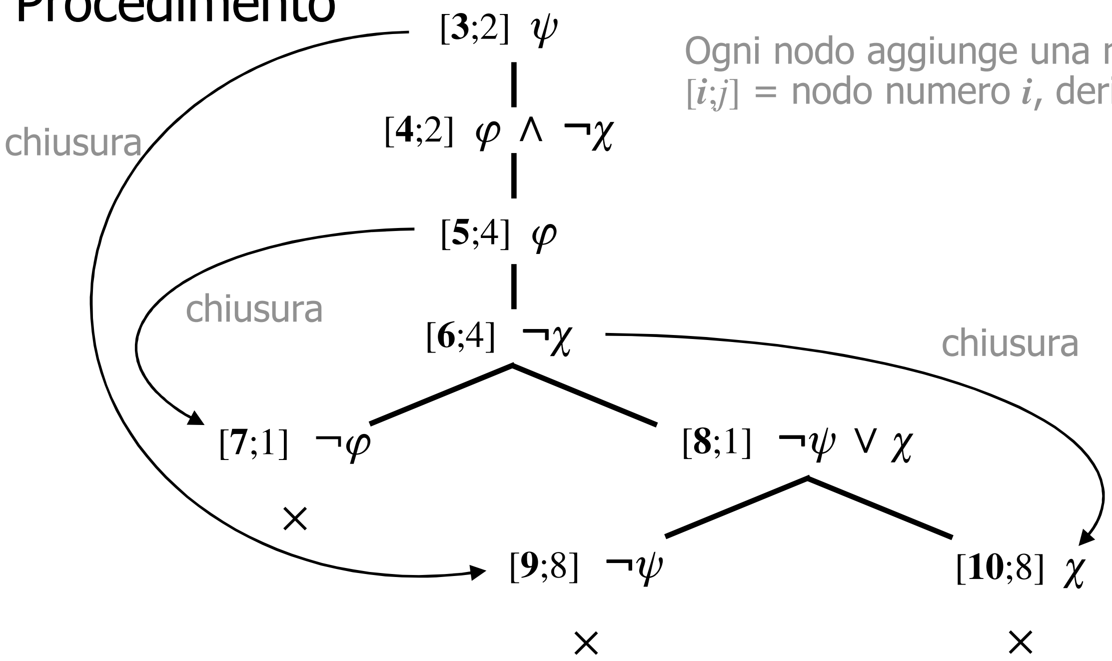
$\varphi \to (\psi \to \chi) \models \psi \to (\varphi \to \chi)$

- Formule di partenza

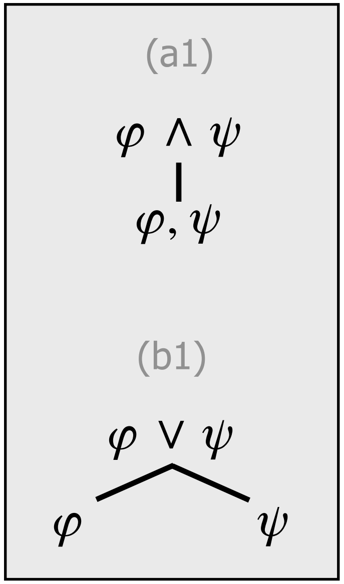
$$(\mathbf{1})\quad \neg\varphi \vee (\neg\psi \vee \chi)$$

$$(\mathbf{2})\quad \psi \wedge (\varphi \wedge \neg\chi)$$

- Procedimento



Ogni nodo aggiunge una nuova formula:
$[i;j]$ = nodo numero i, deriva da j



Regole per i tableau in L_{PO}

- Schema generale delle regole α e β (identiche al caso proposizionale)
  - Le regole α e β rimangono valide per le fbf prive di quantificatori

α	$\alpha_1, \ldots, \alpha_n$
$\phi_1 \wedge \ldots \wedge \phi_n$	$\phi_1, \ldots, \phi_n$
$\neg(\phi_1 \vee \ldots \vee \phi_n)$	$\neg\phi_1, \ldots, \neg\phi_n$
$\neg\neg\phi$	ϕ

β	$\beta_1, \ldots, \beta_n$
$\phi_1 \vee \ldots \vee \phi_n$	$\phi_1, \ldots, \phi_n$
$\neg(\phi_1 \wedge \ldots \wedge \phi_n)$	$\neg\phi_1, \ldots, \neg\phi_n$

$$\begin{array}{c} \alpha \\ \hline \alpha_1 \\ \vdots \\ \alpha_n \end{array}$$

$$\begin{array}{c|c|c} \multicolumn{3}{c}{\beta} \\ \hline \beta_1 & \cdots & \beta_n \end{array}$$

Regole per i tableau in L_{PO} (2)

- Schema generale delle regole γ e δ (eliminazione dei quantificatori)
  - Si assume che le fbf siano in **forma normale prenessa (FNP)**

γ	γ_1
$(\forall x)(\phi(x))$	$\phi(x)$
$\neg(\exists x)(\phi(x))$	$\neg\phi(x)$

δ	δ_1
$\neg(\forall x)(\phi(x))$	$\neg\phi(x)$
$(\exists x)(\phi(x))$	$\phi(x)$

$$\frac{\gamma(x)}{\gamma_1(y)}$$

$y \in \mathrm{Var}$ is new to the tableau.

$$\frac{\delta(x)}{\delta_1(sko_\delta(x_1, \ldots, x_n))}$$

$x_1, \ldots, x_n$ are the free variables in δ.

Algoritmo per i tableau in L_{PO} (*versione incompleta)

- Procedura per $\Gamma \models \varphi$ in logica del primo ordine
  1) Si definisce $\Gamma \cup \{\neg\varphi\}$ come nodo (*tableau*) iniziale
  2) Per ogni nodo, in modalità *depth-first*:
  3) Se possibile, chiudere il nodo **unificando*** una coppia di fbf atomiche complementari
     - se non vi sono altri nodi aperti, la procedura è terminata (successo)
  4) Se il nodo contiene formule composite:
     - applicare le regole α, δ e γ
     - applicare le regole β
  5) Se non si possono applicare regole ed il nodo non è chiuso, la procedura è terminata (fallimento)

***unificazione**

- tramite **MGU**, come nel metodo a risoluzione
- Attenzione: il **MGU** va applicato a <u>tutto</u> il tableau

Esempio 16

Dimostrare che:

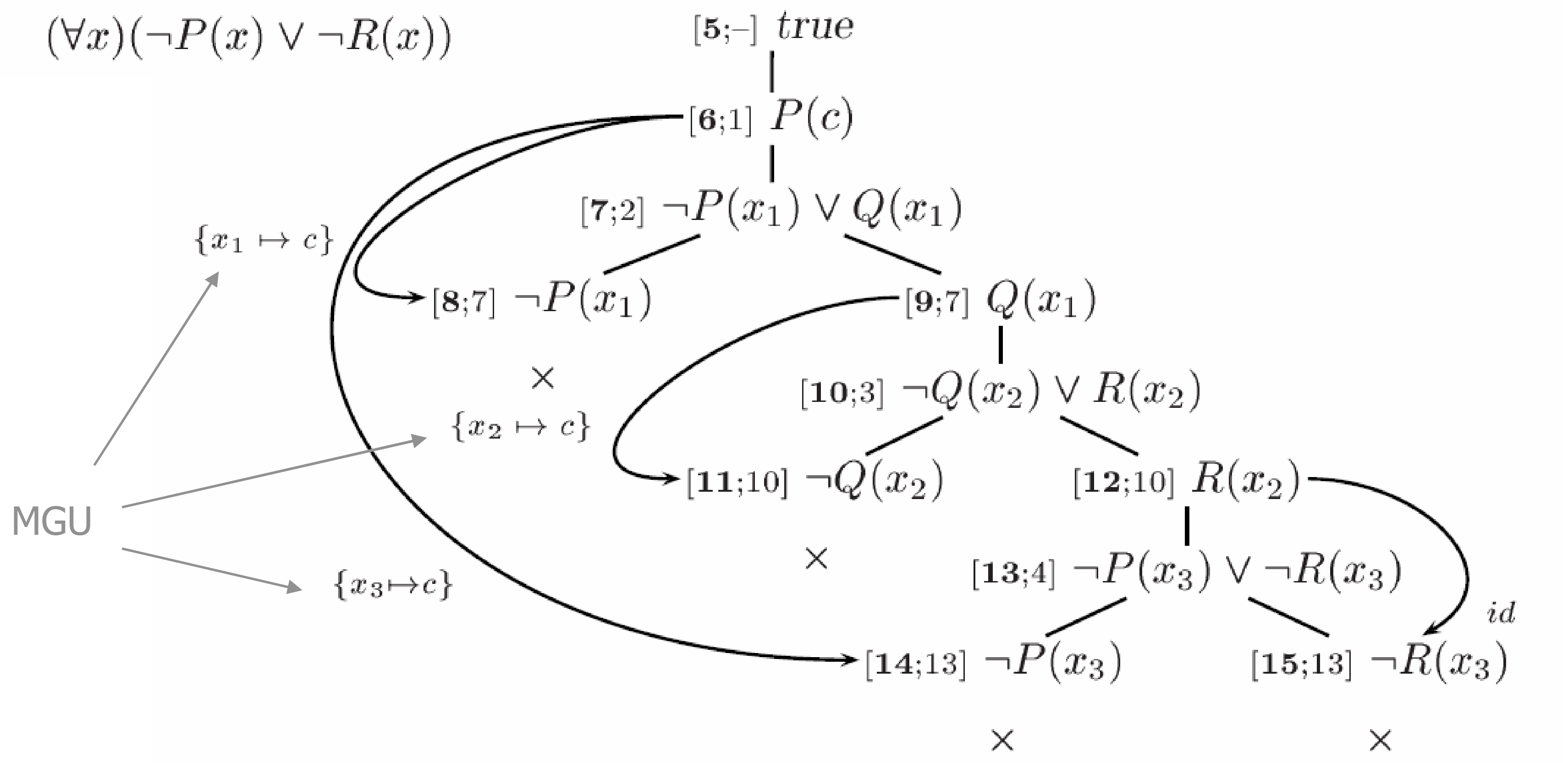
$\exists x\, P(x), \forall x\, (P(x) \rightarrow Q(x)), \forall x\, (Q(x) \rightarrow R(x)) \models \exists x\, (Q(x) \wedge R(x))$

(1) $(\exists x)P(x)$

(2) $(\forall x)(\neg P(x) \vee Q(x))$

(3) $(\forall x)(\neg Q(x) \vee R(x))$

(4) $(\forall x)(\neg P(x) \vee \neg R(x))$



Esempio 17

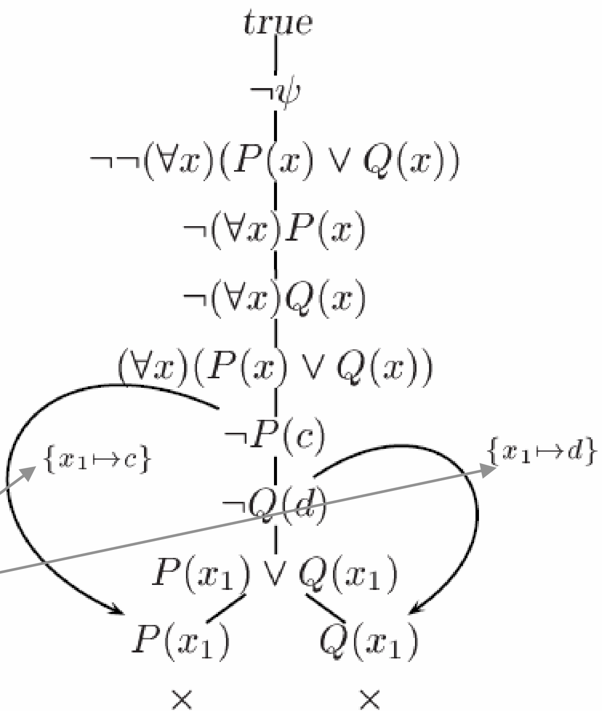
- Il metodo a tableau è **corretto** (se si rispettano le regole)
- Esempio NON CORRETTO:
  le unificazioni sono applicate localmente e non a tutto l'albero

Problema:

$\models \forall x\, (P(x) \vee Q(x)) \rightarrow (\forall x\, P(x) \vee \forall x\, Q(x))$?

(NON è una fbf valida)

Due MGU per
la stessa variabile



Procedura e completezza

- L'applicazione indiscriminata di γ può produrre divergenza

$$\Phi = \{Q \wedge \neg Q, (\forall x)P(x)\}$$

$$\begin{array}{c} true \\ | \\ P(x_1) \\ | \\ P(x_2) \\ \vdots \end{array}$$

γ-rule

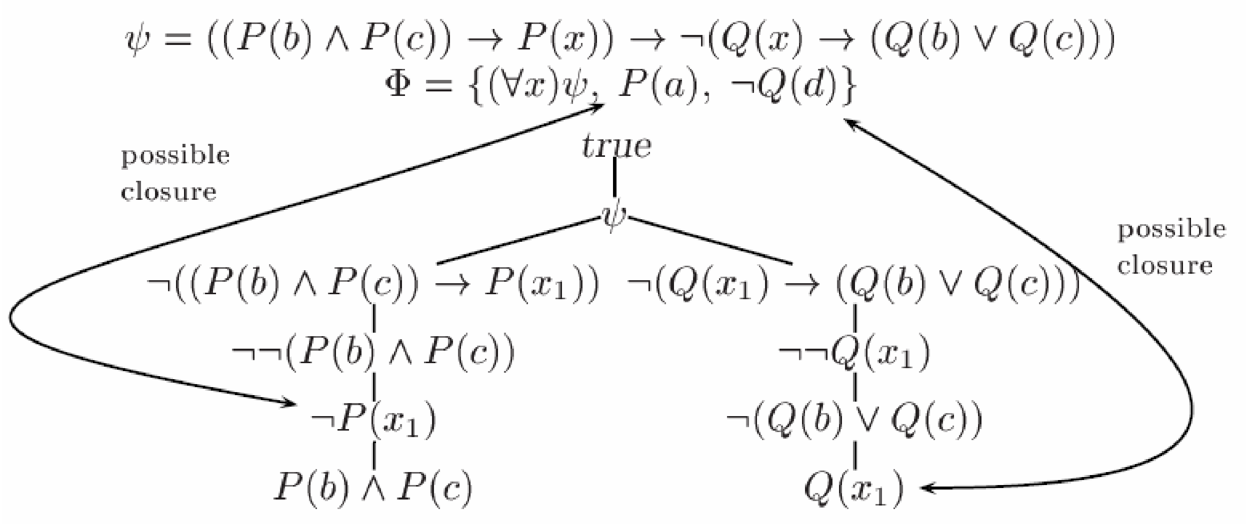
γ-rule

  - Soluzione: applicare le regole secondo un ordine di precedenza
    1) Regole α
    2) Regole δ
    3) Regole β
    4) Regole γ
    - Solo la minore precedenza di γ è collegata alla completezza, il resto è una questione di efficienza

Procedura e completezza (2)

- La chiusura anticipata dei rami può impedire il successo
  - La fbf $\forall x\,(((P(b) \wedge P(c)) \rightarrow P(x)) \rightarrow \neg(Q(x) \rightarrow (Q(b) \vee Q(c))))$ (nell'esempio qui sotto è indicata come $(\forall x)\,\psi$)

    è insoddisfacibile, così come qualsiasi insieme Φ che la contiene



Comunque si scelga tra le due possibilità, si chiude **troppo presto** e si 'consuma' inutilmente la variabile x_1, facendo fallire il metodo

Algoritmo per i tableau in L_{PO} (una versione completa)

- Procedura per $\Gamma \models \varphi$ in logica del primo ordine
  1) Si definisce $\Gamma \cup \{\neg\varphi\}$ come nodo (*tableau*) iniziale
  2) Per ogni nodo, in modalità *breadth-first*:
  3) Identificare tutte le possibili chiusure
     ed aggiungerle ad una lista, con le relative **unificazioni** (*delayed closure*)
  4) Se esiste nella lista un insieme di possibili chiusure (compatibili)
     che chiude l'intero albero, la procedura è terminata (successo)
  5) Applicare una regola, nell'ordine di priorità:
     1) Regola α
     2) Regola δ
     3) Regola β
     4) Regola γ
  6) Se non si possono applicare ulteriori regole
     la procedura è terminata (fallimento)

Considerazioni generali

- Il metodo a tableau per L_{PO}
  - Non è particolarmente efficiente
    - In generale il metodo di risoluzione, ove applicabile, è più efficiente
    - La ricerca in questo settore, tuttavia, è ancora aperta
- Tuttavia:
  - Il metodo a tableau è generalizzabile
    - Pressoché tutte le logiche che tratteremo in seguito ammettono un metodo a tableau
    - Il metodo a risoluzione è invece più strettamente legato a L_{PO}