DEDUZIONE AUTOMATICA IN LOGICA DEI PREDICATI
Sostituzioni e unificazione

Una sostituzione è un insieme finito della forma:

$$\{t_1/x_1, ..., t_n/x_n\}$$

dove:

- $x_1, ..., x_n$ sono variabili
- $t_1, ..., t_n$ sono termini
- per ogni i, $t_i \neq x_i$
- per ogni i, j se $i \neq j$ allora $x_i \neq x_j$

Se E è un'espressione e θ una sostituzione, $\theta = \{t_1/x_1, ..., t_n/x_n\}$, $E\theta$ è l'**istanza** di E che si ottiene sostituendo **simultaneamente** ogni occorrenza di x_i con t_i (per $1 \leq i \leq n$).

Esempio:

$$\text{se } \theta = \{f(z,z)/x, c/z\}$$

$$p(f(x,y), x, g(z))\theta = p(f(f(z,z),y), f(z,z), g(c))$$

Composizione di sostituzioni

Se $\theta = \{t_1/x_1, ..., t_n/x_n\}$,
e $\sigma = \{u_1/y_1, ..., u_m/y_m\}$,
allora

$$\theta \circ \sigma$$

si ottiene da

$$\{t_1\sigma/x_1, ..., t_n\sigma/x_n, u_1/y_1, ..., u_m/y_m\}$$

eliminando tutti gli elementi:

$$t_j\sigma/x_j \text{ tali che } t_j\sigma = x_j$$
$$u_i/y_i \text{ tali che } y_i \in \{x_1, ..., x_n\}$$

Composizione di sostituzioni: esempio

Se

$$\theta = \{f(y)/x, z/y\}$$
$$\sigma = \{a/x, b/y, y/z\}$$

Allora, $\theta \circ \sigma$ si ottiene da

$$\{f(b)/x, y/y, a/x, b/y, y/z\}$$

eliminando $a/x, b/y, y/y$:

$$\theta \circ \sigma = \{f(b)/x, y/z\}$$

Proprietà fondamentale della composizione di sostituzioni

Per ogni espressione E:

$$E(\theta \circ \sigma) \;=\; (E\theta)\sigma$$

Esempio: se θ e σ sono come sopra:

$$h(x, g(y), z)(\theta \circ \sigma) = h(x, g(y), z)\{f(b)/x, y/z\} = h(f(b), g(y), y)$$

$$(h(x, g(y), z)\theta)\sigma = h(f(y), g(z), z)\sigma = h(f(b), g(y), y)$$

Unificazione

Una sostituzione θ è un **unificatore** per l'insieme di espressioni $\{E_1, ..., E_k\}$ sse:

$$E_1\theta = E_2\theta = ... = E_k\theta$$

L'insieme $\{E_1, ..., E_k\}$ è unificabile sse esiste un unificatore per esso

Unificatore più generale

Un unificatore σ per $\{E_1, ..., E_k\}$ è un

unificatore più generale
(**mgu** = most general unifier)

sse per ogni unificatore θ di $\{E_1, ..., E_k\}$ esiste una sostituzione λ tale che $\theta = \sigma \circ \lambda$.

Cioè ogni altro unificatore θ dell'insieme di espressioni "sostituisce di più" di σ: per ogni espressione E

$$E\theta = (E\sigma)\lambda$$

Esempio

$\theta = \{f(a)/x, a/y\}$ è un unificatore per

$$\{p(x), p(f(y))\}$$

ma non è un mgu.

Infatti per "unificare" $p(x)$ con $p(f(y))$ è sufficiente sostituire x con $f(y)$:

$$\sigma = \{f(y)/x\} \text{ è un mgu di } \{p(x), p(f(y))\}$$

$\theta = \sigma \circ \{a/y\}$;
d'altra parte non esiste alcuna sostituzione λ tale che $\sigma = \theta \circ \lambda$.

Se σ è un mgu per $\{E_1, ..., E_k\}$, vuol dire che:

- σ è un unificatore:
$$E_1\sigma = E_2\sigma = ... = E_k\sigma$$
- è più generale: se
$$E_1\theta = E_2\theta = ... = E_k\theta$$
allora per qualche λ
$$E_i\theta = (E_i\sigma)\lambda$$
(θ è meno generale: sostituisce più del necessario)

Algoritmo di unificazione di Robinson
per due espressioni

Definizione. Siano A e B due espressioni e sia k la posizione più a sinistra in cui le due sequenze di simboli differiscono. L'insieme $\{e_1, e_2\}$ delle due sottoespressioni che iniziano alla posizione k in A e B si chiama il ***disagreement set*** di A e B.

Esempio: se $A = p(x, f(c), x)$, $B = p(x, g(y, a), z)$, il disagreement set di A e B è:

$$\{f(c), g(y, a)\}$$

Algoritmo: calcola $mgu(A, B)$

- Inizializzazione:

$$\begin{array}{ll} A_0 = A & \sigma_0 = \emptyset \\ B_0 = B & i = 0 \end{array}$$

- Ciclo:
  - Se $A_i = B_i$: uscire dal ciclo, con risultato σ_i, altrimenti proseguire
  - Sia $\{e, e'\}$ il disagreement set di A_i e B_i. Consideriamo i seguenti casi:

    1. se un elemento di $\{e, e'\}$ è una variabile x_i e l'altro un'espressione e_i in cui non occorre x_i, allora porre:

    $$\begin{aligned} \sigma_{i+1} &= \sigma_i \circ \{e_i/x_i\} \\ A_{i+1} &= A_i\{e_i/x_i\} \\ B_{i+1} &= B_i\{e_i/x_i\} \end{aligned}$$

       e proseguire nel ciclo.
    2. altrimenti uscire dal ciclo e riportare un fallimento: $\{e, e'\}$ non è unificabile

Esercizi

Si mostrino l'esecuzione, stadio per stadio, ed il risultato dell'algoritmo di unificazione di Robinson sulle seguenti coppie di espressioni:

1. $A = p(x)$, $B = p(f(y))$
2. $A = p(x, f(a))$, $B = p(c, f(x))$
3. $A = p(x, f(a))$, $B = p(c, f(y))$
4. $A = p(x, x)$, $B = p(f(y), y)$
5. $A = p(x, f(c), x)$, $B = p(f(y), y, z)$
6. $A = p(x, f(c), x)$, $B = p(f(y), y, c)$

Deduzione automatica: forward chaining

Assumiamo che la KB sia composta da **clausole definite**:

$$p_1 \wedge ... \wedge p_n \rightarrow p$$

con p_i e p atomiche. Le formule sono intese come quantificate universalmente:

$$\forall x_1 ... \forall x_k (p_1 \wedge ... \wedge p_n \rightarrow p)$$

La regola di inferenza per la deduzione in avanti è una specie di *modus ponens* generalizzato:

$$\frac{p'_1 \qquad ... \qquad p'_n \qquad p_1 \wedge ... \wedge p_n \rightarrow p}{p\theta}$$

$$\text{se } \theta = mgu(p_1 \wedge ... \wedge p_n, p'_1 \wedge ... \wedge p'_n)$$

Esempio

KB: $antenato(z, y) \wedge figlio(x, y) \rightarrow antenato(z, x)$

$figlio(x, y) \rightarrow antenato(y, x)$

$antenato(padre(ernesto), paolo)$

TELL: $figlio(pietro, paolo)$

$$\frac{figlio(pietro, paolo)\ ; \quad figlio(x, y) \rightarrow antenato(y, x)}{antenato(paolo, pietro)}$$

$$\theta = \{pietro/x, paolo/y\}$$

$$figlio(pietro, paolo)\theta = figlio(x, y)\theta$$

$$\frac{\begin{array}{c} antenato(padre(ernesto), paolo)\ ; \ \ figlio(pietro, paolo)\ ; \\ antenato(z, y) \wedge figlio(x, y) \rightarrow antenato(z, x) \end{array}}{antenato(padre(ernesto), pietro)}$$

$$\theta = \{padre(ernesto)/z, paolo/y, pietro/x\}$$

$$\begin{array}{c} antenato(padre(ernesto), paolo)\theta = antenato(z, y)\theta \\ figlio(pietro, paolo)\theta = figlio(x, y)\theta \end{array}$$

Deduzione automatica: backward chaining

L'unificazione è utilizzata praticamente in tutti i metodi di deduzione per la logica dei predicati

- Tableaux per la logica dei predicati
- Risoluzione