Intelligenza Artificiale I

Esercitazione 3

Marco Piastra

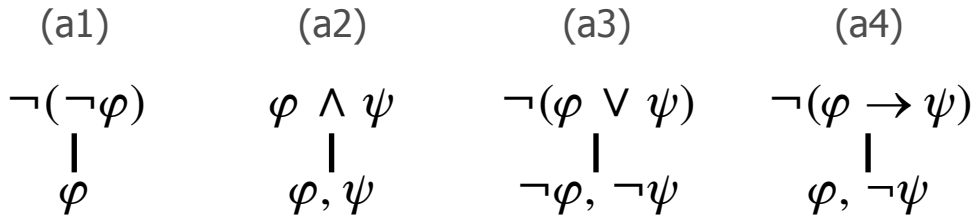
Semantic Tableau, regole alfa e beta

- Un tableau è un'insieme di fbf

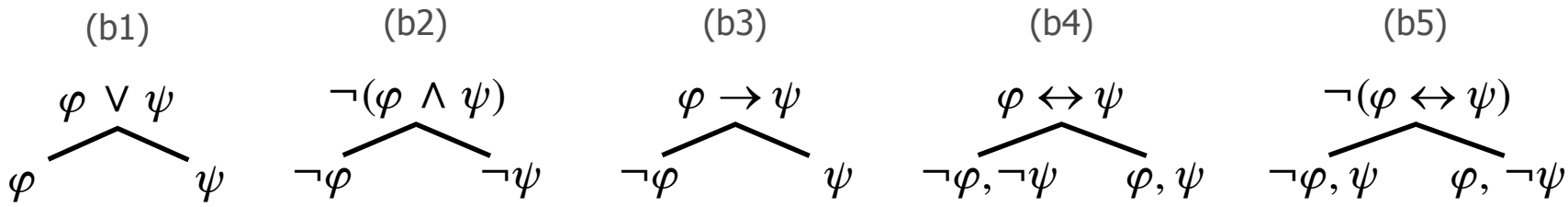
  Le regole (alfa e beta) trasformano un tableau in uno o due nuovi tableau

  Si parte da un tableau iniziale

- Regole alfa (o di espansione)



- Regole beta (o di biforcazione)



Metodo dei *Semantic Tableau*

- Procedura:

  Dato il problema $\Gamma \models \varphi$

  Si assume $\Gamma \cup \{\neg\varphi\}$ come nodo *(tableau)* iniziale

  Su ciascun ramo, in modalità *depth-first*

  Se il nodo contiene solo letterali,

  se il nodo contiene una contraddizione, chiudere il ramo
  altrimenti terminare la procedura [fallimento]

  Se il nodo contiene formule composite:

  a) applicare le regole alfa
  b) applicare le regole beta

  Notare: le regole alfa vengono applicate prima delle regole beta

deffunction

(file semanticTableau.jess)

- Definisce una funzione

Esempio:

```
(deffunction duplicate-deep (?n)
    (if (symbolp ?n)
        then
        (return ?n)
        else
        (return
            (duplicate ?n
                (id (gensym*))  ;; Always make sure it has a unique id
                (op1 (duplicate-deep (fact-slot-value ?n op1)))
                (op2 (duplicate-deep (fact-slot-value ?n op2)))
            )
        )
    )
)
```

deffunction

(file semanticTableau.jess)

- Definisce una funzione

nome

Esempio:

```
(deffunction duplicate-deep (?n)
    (if (symbolp ?n)
        then
        (return ?n)
        else
        (return
            (duplicate ?n
                (id (gensym*))  ;; Always make sure it has a unique id
                (op1 (duplicate-deep (fact-slot-value ?n op1)))
                (op2 (duplicate-deep (fact-slot-value ?n op2)))
            )
        )
    )
)
```

deffunction

(file semanticTableau.jess)

- Definisce una funzione

parametri

Esempio:

```
(deffunction duplicate-deep (?n)
    (if (symbolp ?n)
        then
        (return ?n)
        else
        (return
            (duplicate ?n
                (id (gensym*))  ;; Always make sure it has a unique id
                (op1 (duplicate-deep (fact-slot-value ?n op1)))
                (op2 (duplicate-deep (fact-slot-value ?n op2)))
            )
        )
    )
)
```

deffunction

(file semanticTableau.jess)

- Definisce una funzione

Esempio:

```
(deffunction duplicate-deep (?n)
    (if (symbolp ?n)
        then
        (return ?n)
        else
        (return
            (duplicate ?n
                (id (gensym*))  ;; Always make sure it has a unique id
                (op1 (duplicate-deep (fact-slot-value ?n op1)))
                (op2 (duplicate-deep (fact-slot-value ?n op2)))
            )
        )
    )
)
```

controllo di flusso

deffunction

(file semanticTableau.jess)

- Definisce una funzione

Esempio:

```
(deffunction duplicate-deep (?n)
    (if (symbolp ?n)
        then
        (return ?n)
        else
        (return
            (duplicate ?n
                (id (gensym*))  ;; Always make sure it has a unique id
                (op1 (duplicate-deep (fact-slot-value ?n op1)))
                (op2 (duplicate-deep (fact-slot-value ?n op2)))
            )
        )
    )
)
```

ritorno di valori

deffunction

(file semanticTableau.jess)

- Definisce una funzione

Esempio:

```
(deffunction duplicate-deep (?n)
    (if (symbolp ?n)
        then
        (return ?n)
        else
        (return
            (duplicate ?n
                (id (gensym*))  ;; Always make sure it has a unique id
                (op1 (duplicate-deep (fact-slot-value ?n op1)))
                (op2 (duplicate-deep (fact-slot-value ?n op2)))
            )
        )
    )
)
```

predicato speciale di Jess:
vale TRUE se l'argomento è un *simbolo*

deffunction

(file semanticTableau.jess)

- Definisce una funzione

Esempio:

```
(deffunction duplicate-deep (?n)
    (if (symbolp ?n)
        then
        (return ?n)
        else
        (return
            (duplicate ?n
                (id (gensym*))  ;; Always make sure it has a unique id
                (op1 (duplicate-deep (fact-slot-value ?n op1)))
                (op2 (duplicate-deep (fact-slot-value ?n op2)))
            )
        )
    )
)
```

funzione speciale di Jess:
ritorna un (nuovo) *simbolo* univoco

Semantic Tableau

(file semanticTableau.jess)

- Domande:
  a) Cosa vuol dire metodo per *refutazione*?
     (non procedere oltre prima di aver capito)
  b) Come vengono rappresentate le formule?
  c) Come vengono rappresentati i tableau?
  d) Come funziona l'implementazione Jess del metodo?
  e) Perchè si usa **`duplicate-deep`** e **`retract-deep`**?
  f) A cosa servono le priorità?

(usare l'esempio st-TheProblem.jess)