

# INTELLIGENZA ARTIFICIALE

A N N O   A C C A D E M I C O  
2 0 0 8 - 2 0 0 9

*La Rappresentazione della Conoscenza:  
I Sistemi di Produzione*

# SOMMARIO

- Formalismi dichiarativi e procedurali
- Le regole di produzione
- Sistemi di produzioni forward chaining
- Sistemi di produzioni backward chaining

# FORMALISMI DICHIARATIVI E PROCEDURALI

- ✿ Nell'IA si distingue fra formalismi di rappresentazione di tipo **dichiarativo** e di tipo **procedurale**.
- ✿ La logica è un tipico formalismo dichiarativo.
- ✿ Anche le reti semantiche, che vedremo successivamente, sono un formalismo dichiarativo.
- ✿ Ciò significa semplicemente che mediante un formalismo di questo tipo è possibile descrivere un dominio, utilizzando poi la descrizione in modi diversi per risolvere problemi di differente natura.

## FORMALISMO PROCEDURALE

- ✻ In molti casi si desidera fornire a un sistema la conoscenza, unitamente a una *prescrizione precisa* sul modo di utilizzarla.
- ✻ In altre parole, non si desidera tanto descrivere un dominio, quanto specificare bene come il sistema dovrà comportarsi in determinate situazioni.
- ✻ In questo caso si preferisce l'uso di un formalismo di tipo procedurale, come i **sistemi di produzioni**.

# LE REGOLE DI PRODUZIONE

- ✿ Alla base dei sistemi di produzioni sta il concetto di **regola di produzione**, ossia di regole del tipo:

SE  $x$  ALLORA  $y$

- ✿ Come gli operatori dello spazio degli stati, le regole di produzione possono essere utilizzate in avanti (**forward**) o all'indietro (**backward**).
- ✿ Poiché i due casi presentano differenze notevoli, ce ne occuperemo separatamente.

# I SISTEMI DI PRODUZIONI FORWARD CHAINING

- ✿ I sistemi di produzioni in avanti (**forward chaining**) si ispirano a un'idea del matematico Post (1943), che se ne servì per indagare le basi teoriche del concetto di calcolo.
- ✿ Oggi tali sistemi possono essere considerati dei veri e propri linguaggi di programmazione speciali per l'IA.
- ✿ In genere, essi sono costituiti da tre componenti:
  - ✿ la **base delle regole**, che contiene tutte le regole di produzione del sistema;
  - ✿ La **memoria di lavoro**, che serve a memorizzare lo stato raggiunto nel processo di risoluzione di un problema;
  - ✿ l'**interprete**, che utilizza le regole e la memoria per realizzare il processo di risoluzione automatica di un problema.

# I SISTEMI DI PRODUZIONI FORWARD CHAINING

- ☼ Come visto in precedenza, le regole di produzione hanno la forma:

SE **condizione** x ALLORA **azione** y

- ☼ Per **condizione** si intende una lista di **fatti**; se tutti i fatti contenuti nella condizione di una regola sono presenti nella memoria di lavoro, si dice che la regola è **soddisfatta**.

# I SISTEMI DI PRODUZIONI FORWARD CHAINING

Ad esempio, la regola:

**SE** luogo lontano **E** luogo in città  
**ALLORA** + prendere mezzo pubblico

è soddisfatta quando la memoria di lavoro contiene, fra gli altri, i seguenti fatti:

- luogo-lontano
- luogo-in-città



# I SISTEMI DI PRODUZIONI FORWARD CHAINING

- ✿ Quando una regola è soddisfatta, può essere **attivata** dall'interprete.
- ✿ Attivare una regola significa eseguire le azioni specificate dopo la parola **ALLORA**.
- ✿ Nella regola vista prima, abbiamo supposto che il simbolo **+** significhi: “aggiungere il fatto indicato alla memoria di lavoro”.
- ✿ Se la regola viene attivata, la memoria di lavoro verrà modificata aggiungendo: “**prendere mezzo pubblico**”.

# I SISTEMI DI PRODUZIONI FORWARD CHAINING

## Esempio

Quella che segue è una piccola base di regole per scegliere il mezzo di trasporto più appropriato a una situazione:

- R1: SE luogo-lontano ALLORA + andare-in-automobile
- R2: SE luogo-vicino ALLORA + andare-a-piedi
- R3: SE luogo-lontano E luogo-in-città  
ALLORA + prendere-mezzo-pubblico
- R4: SE prendere-mezzo-pubblico ALLORA + andare-in-bus
- R5: SE prendere-mezzo-pubblico E molta-fretta  
ALLORA + andare-in-taxi
- R6: SE andare-in-automobile ALLORA → in-automobile E STOP
- R7: SE andare-a-piedi ALLORA → a-piedi E STOP
- R8: SE andare-in-bus ALLORA → in-bus E STOP
- R9: SE andare-in-taxi ALLORA → in-taxi E STOP

# I SISTEMI DI PRODUZIONI FORWARD CHAINING

## Esempio

Supponiamo che ci interessi recarci in fretta in un luogo lontano della città. Per risolvere questo problema dobbiamo innanzitutto caricare nella memoria di lavoro la descrizione del problema:

- luogo-lontano
- luogo-in-città
- molta-fretta

# I SISTEMI DI PRODUZIONI FORWARD CHAINING

## Esempio

A questo punto avviamo l'interprete, che si comporta nel modo seguente:

- si controllano tutte le regole contenute nella base, identificando quelle che sono soddisfatte dalla memoria di lavoro attuale; in questo caso si tratta di R1 e R3;
- tra le regole identificate, se ne sceglie una per l'attivazione; in questo caso è opportuno scegliere R3, perché è più **specific**a, in quanto la sua condizione corrisponde a due fatti contenuti nella memoria, mentre la condizione di R1 corrisponde a un solo fatto;

# I SISTEMI DI PRODUZIONI FORWARD CHAINING

## Esempio

- si attiva la regola prescelta, cioè R3, e si esegue l'azione corrispondente, aggiungendo nella memoria di lavoro il fatto: “prendere-mezzo-pubblico”;
- si riprende da capo, controllando le regole contenute nella base rispetto al nuovo stato della memoria di lavoro; poiché nessun fatto è stato tolto dalla memoria, R1 e R3 sono ancora soddisfatte; in più, sono soddisfatte R4 e R5;

# I SISTEMI DI PRODUZIONI FORWARD CHAINING

## Esempio

- ora si sceglie la regola da attivare; R3 viene subito scartata in quanto è già stata usata, e non può quindi aggiungere nuova informazione nella memoria; fra le altre, viene scelta R5, perché è più specifica;
- R5 viene attivata, aggiungendo alla memoria il fatto: “andare-in-taxi”;
- si riprende ancora da capo, controllando le regole rispetto al nuovo stato della memoria di lavoro; sono soddisfatte R1, R3, R4, R5 e R9;

# I SISTEMI DI PRODUZIONI FORWARD CHAINING

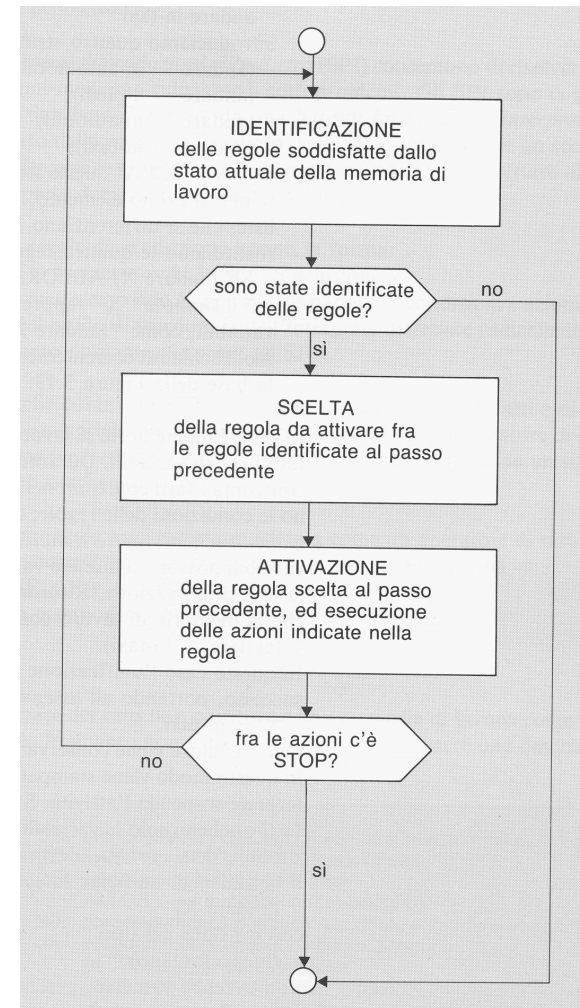
## Esempio

- le regole R3 e R5 vengono scartate, perché sono già state attivate in passato; fra R1, R4 e R9 scegliamo R9 perché è soddisfatta da un fatto inserito nella memoria **più recentemente** degli altri; questo criterio è utilizzato spesso in tali sistemi e assicura che le informazioni acquisite di recente vengano prese in considerazione prima delle informazioni “vecchie”;
- viene attivata R9, che specifica due azioni: la visualizzazione della soluzione (**→**) e la fermata dell'interprete (**STOP**).

# I SISTEMI DI PRODUZIONI FORWARD CHAINING

L'esempio appena illustrato, benché semplicissimo, mostra nelle sue linee essenziali il funzionamento di un sistema di produzioni forward chaining.

Nella figura qui a fianco è schematizzato il modo di funzionare dell'interprete, basato sul ciclo di **identificazione**, **scelta** e **attivazione** delle regole.





# I SISTEMI DI PRODUZIONI FORWARD CHAINING

- ✿ In genere, gli interpreti di sistemi forward chaining differiscono fra loro soprattutto rispetto ai criteri utilizzati nella scelta della regola da attivare.
- ✿ I criteri visti nell'esempio sono tra i più comuni:
  - ✿ non attivare più di una volta la stessa regola, o comunque non attivare una regola che non porterebbe nessuna modifica allo stato della memoria;
  - ✿ attivare sempre la regola **più specifica** fra tutte quelle soddisfatte dallo stato della memoria;
  - ✿ preferire le regole soddisfatte dai fatti che sono stati introdotti in memoria **più di recente**.

# I SISTEMI DI PRODUZIONI FORWARD CHAINING

- ✿ Di fronte a problemi complessi, tuttavia, può essere necessario escogitare criteri di scelta più specifici.
- ✿ Nel nostro esempio la struttura dei fatti contenuti nella memoria di lavoro è semplice: ogni fatto è rappresentato da un simbolo unico, come “luogo-lontano” e “andare-in-pensione”.
- ✿ In genere i fatti hanno una forma più complessa, e sono rappresentati da strutture simboliche costituite da più simboli, fra cui **variabili**.

# I SISTEMI DI PRODUZIONI FORWARD CHAINING

- ✿ Per avere un'idea dell'uso delle variabili, si considerino le regole R6-R9 viste in precedenza:

R6: SE andare-in-automobile ALLORA → in-automobile E STOP

R7: SE andare-a-piedi ALLORA → a-piedi E STOP

R8: SE andare-in-bus ALLORA → in-bus E STOP

R9: SE andare-in-taxi ALLORA → in-taxi E STOP

- ✿ Queste regole hanno una forma molto simile, e possiamo ridurle a una regola unica agendo nel modo seguente:

Al posto dei quattro simboli distinti:

andare-in-automobile

andare-a-piedi

andare-in-bus

andare-in-taxi

introduciamo quattro strutture simboliche:

(andare in-automobile)

(andare a-piedi)

(andare in-bus)

(andare in-taxi)

# I SISTEMI DI PRODUZIONI FORWARD CHAINING

- ✿ Sostituiamo le quattro regole R6-R9 con l'unica regola:

SE (andare X) ALLORA → X E STOP

- ✿ Qui il simbolo X rappresenta una variabile.
- ✿ Le condizioni contenenti variabili, come “(andare X)”, sono detti **schemi** (pattern).

# I SISTEMI DI PRODUZIONI FORWARD CHAINING

## Esempio

Modifichiamo coerentemente le altre regole, ottenendo alla fine la seguente base:

R1: SE luogo-lontano ALLORA + (andare in-automobile)

R2: SE luogo-vicino ALLORA + (andare a-piedi)

R3: SE luogo-lontano E luogo-in-città  
ALLORA + prendere-mezzo-pubblico

R4: SE prendere-mezzo-pubblico ALLORA + (andare in-bus)

R5: SE prendere-mezzo-pubblico E molta-fretta  
ALLORA + (andare in-taxi)

R6: SE (andare X) ALLORA → X E STOP

# I SISTEMI DI PRODUZIONI FORWARD CHAINING

- ✿ In genere, la memoria di lavoro non può contenere pattern, ma solo fatti privi di variabili.
- ✿ Durante la fase di identificazione, l'interprete confronta i fatti contenuti nella memoria con i pattern che formano le condizioni delle regole.
- ✿ Questo confronto viene detto **pattern matching**.

# I SISTEMI DI PRODUZIONI FORWARD CHAINING

- ✻ Nel pattern matching le variabili possono assumere valori arbitrari, che conservano fino alla fase di attivazione della regola relativa.
- ✻ Ritornando al nostro esempio, supponiamo che la memoria di lavoro contenga il fatto: “**(andare in-taxi)**”.
- ✻ In questo caso il pattern matching con la condizione della regola R6 ha successo, portando all’assegnamento del valore **X = in-taxi**.
- ✻ Questo valore viene conservato nella successiva fase di attivazione della regola.

# I SISTEMI DI PRODUZIONI

## FORWARD CHAINING

- ✿ Si noti che quando si tenta di unificare fatti o pattern, solo le variabili possono assumere un valore; gli altri simboli, detti costanti, devono essere uguali due a due.
- ✿ Pertanto il tentativo di unificare lo schema (andare X) con un fatto del tipo (luogo lontano) non avrebbe successo, perché “andare” è un simbolo diverso da “luogo”.



# I SISTEMI DI PRODUZIONI FORWARD CHAINING

- ✻ Un tipico sistema di produzioni forward chaining può eseguire un certo numero di azioni diverse, fra cui:
  - ✻ introdurre un fatto in memoria;
  - ✻ cancellare un fatto dalla memoria;
  - ✻ modificare in parte un fatto già contenuto nella memoria;
  - ✻ gestire l'ingresso e l'uscita di dati da un terminale o dalla memoria di massa.

# I SISTEMI DI PRODUZIONI FORWARD CHAINING

- ✻ Molti sistemi di produzioni forward chaining sono realizzati come estensioni di linguaggi di programmazione più tradizionali.
- ✻ In questo caso, un'azione può consistere anche nell'attivazione di una procedura esterna scritta in quel linguaggio di programmazione.
- ✻ Si può così realizzare un processo di risoluzione di problemi in cui vengono utilizzati anche metodi di calcolo tradizionali quando questi risultino adatti (ad es. per eseguire elaborazioni di tipo numerico come parte di un più complesso procedimento di risoluzione di un problema).

# I SISTEMI DI PRODUZIONI BACKWARD CHAINING

- ✿ I sistemi di produzioni backward chaining consentono di risolvere problemi mediante riduzione a sotto-problemi.
- ✿ Essi sono costituiti da una **base di regole**, da una **base di fatti** e da un **interprete**.
- ✿ Al posto della memoria di lavoro dei sistemi forward chaining, intesa come un semplice insieme di fatti, abbiamo ora un **albero AND/OR**, costruito dall'interprete durante il processo risolutivo.

# I SISTEMI DI PRODUZIONI BACKWARD CHAINING

- ✿ Le **regole di produzione** hanno la forma:

**SE** premesse **x** **ALLORA** conclusione **y**

- ✿ Il questo caso non è appropriato parlare di condizioni e azioni, perché il modo di funzionare dell'interprete differisce radicalmente da quanto abbiamo visto per i sistemi forward chaining.
- ✿ Anche qui le premesse sono costituite da una lista di fatti o di schemi con variabili, eventualmente preceduti da un operatore di negazione (vedi oltre). La conclusione è di solito formata da un singolo fatto o schema.

# I SISTEMI DI PRODUZIONI BACKWARD CHAINING

- ✱ Ogni regola può essere considerata come un operatore di riduzione a sotto-problemi; ad esempio, una regola della forma:

SE  $x$  E  $y$  ALLORA  $z$

può essere vista come un operatore che riduce il problema  $z$  nei due sotto-problemi  $x$  e  $y$ .

# I SISTEMI DI PRODUZIONI BACKWARD CHAINING

- ✻ La base dei fatti non costituisce una memoria di lavoro come nei sistemi forward chaining, in quanto non viene utilizzata per mantenere una descrizione dello stato raggiunto nel processo di risoluzione del problema.
- ✻ Piuttosto, nella base dei fatti vengono rappresentati i problemi primitivi, che possono essere dati per risolti senza ulteriori riduzioni.
- ✻ Come vedremo tra poco in un esempio, la base di fatti descrive anche le particolari condizioni sotto cui un problema deve essere risolto.

# I SISTEMI DI PRODUZIONI BACKWARD CHAINING

## Esempio

Base di regole per risolvere lo stesso problema visto prima nel caso dei sistemi forward chaining:

- R1: SE luogo-lontano E NON luogo-in-città  
    ALLORA (andare in-automobile)
- R2: SE luogo-vicino ALLORA (andare a-piedi)
- R3: SE luogo-lontano E luogo-in-città  
    ALLORA prendere-mezzo-pubblico
- R4: SE prendere-mezzo-pubblico E NON molta-fretta  
    ALLORA (andare in-bus)
- R5: SE prendere-mezzo-pubblico E molta-fretta  
    ALLORA (andare in-taxi)

# I SISTEMI DI PRODUZIONI BACKWARD CHAINING

- ✱ Introduciamo nella base dei fatti le condizioni sotto cui desideriamo risolvere il problema di trovare il mezzo di trasporto più adatto.
- ✱ Questi fatti sono gli stessi che avevamo utilizzato per inizializzare la memoria di lavoro del sistema forward chaining:
  - luogo-lontano
  - luogo-in-città
  - molta-fretta



# I SISTEMI DI PRODUZIONI BACKWARD CHAINING

- ✿ Assegniamo l'obiettivo da raggiungere, ovvero la radice dell'albero AND/OR che l'interprete dovrà costruire:

**(andare X)**

- ✿ In questo caso l'obiettivo è un pattern contenente una variabile.
- ✿ Ciò che ci interessa realmente è proprio il valore che il sistema assegnerà ad **X** nel corso del processo risolutivo; valore che costituisce la soluzione cercata.

# I SISTEMI DI PRODUZIONI BACKWARD CHAINING

- ✱ A questo punto l'interprete identifica tutte le regole la cui conclusione (la parte dopo **ALLORA**) si unifica con l'obiettivo.
- ✱ Nel nostro caso vengono identificate:
  - R1 con l'assegnamento di valore **X = in-automobile**
  - R2 con l'assegnamento di valore **X = a-piedi**
  - R4 con l'assegnamento di valore **X = in-bus**
  - R5 con l'assegnamento di valore **X = in-taxi**

# I SISTEMI DI PRODUZIONI BACKWARD CHAINING

- ✱ Si sceglie una regola da attivare; supponiamo di seguire semplicemente l'ordine delle regole, scegliendo R1.
- ✱ Si attiva R1, operando la riduzione dell'obiettivo a un certo numero di sotto-problemi.
- ✱ Nel nostro caso, l'obiettivo:

**(andare X)**

con l'assegnamento di valore:

**X = in-automobile**

viene sostituito dalle premesse della regola R1:

**luogo-lontano E NON luogo-in-città**

che vengono assunte come nuovo obiettivo da raggiungere.

# I SISTEMI DI PRODUZIONI BACKWARD CHAINING

- ✱ L'interprete seleziona ora uno dei due sotto-problemi e lo affronta per primo.
- ✱ Ancora, supponiamo di rispettare l'ordine, e selezioniamo **“luogo-lontano”** come primo obiettivo da raggiungere.
- ✱ Poiché **“luogo-lontano”** è nella base dei fatti, e va quindi considerato come un problema primitivo, diamo questo obiettivo per raggiunto.

# I SISTEMI DI PRODUZIONI BACKWARD CHAINING

- ✿ Selezioniamo il prossimo sotto-problema da risolvere, ovvero: “**NON luogo-in-città**”.
- ✿ In genere, la negazione viene trattata nel modo seguente: dato un obiettivo del tipo “**NON x**” si affronta l’obiettivo **x**. Se questo viene raggiunto con successo, allora l’obiettivo “**NON x**” viene dato per fallito; se invece l’obiettivo **x** non viene raggiunto, allora “**NON x**” è risolto con successo.
- ✿ Pertanto nel nostro caso affrontiamo l’obiettivo “**luogo-in-città**”.
- ✿ Poiché è nella base dei fatti, esso è immediatamente raggiunto. Ciò comporta il fallimento dell’obiettivo “**NON luogo-in-città**”.

# I SISTEMI DI PRODUZIONI BACKWARD CHAINING

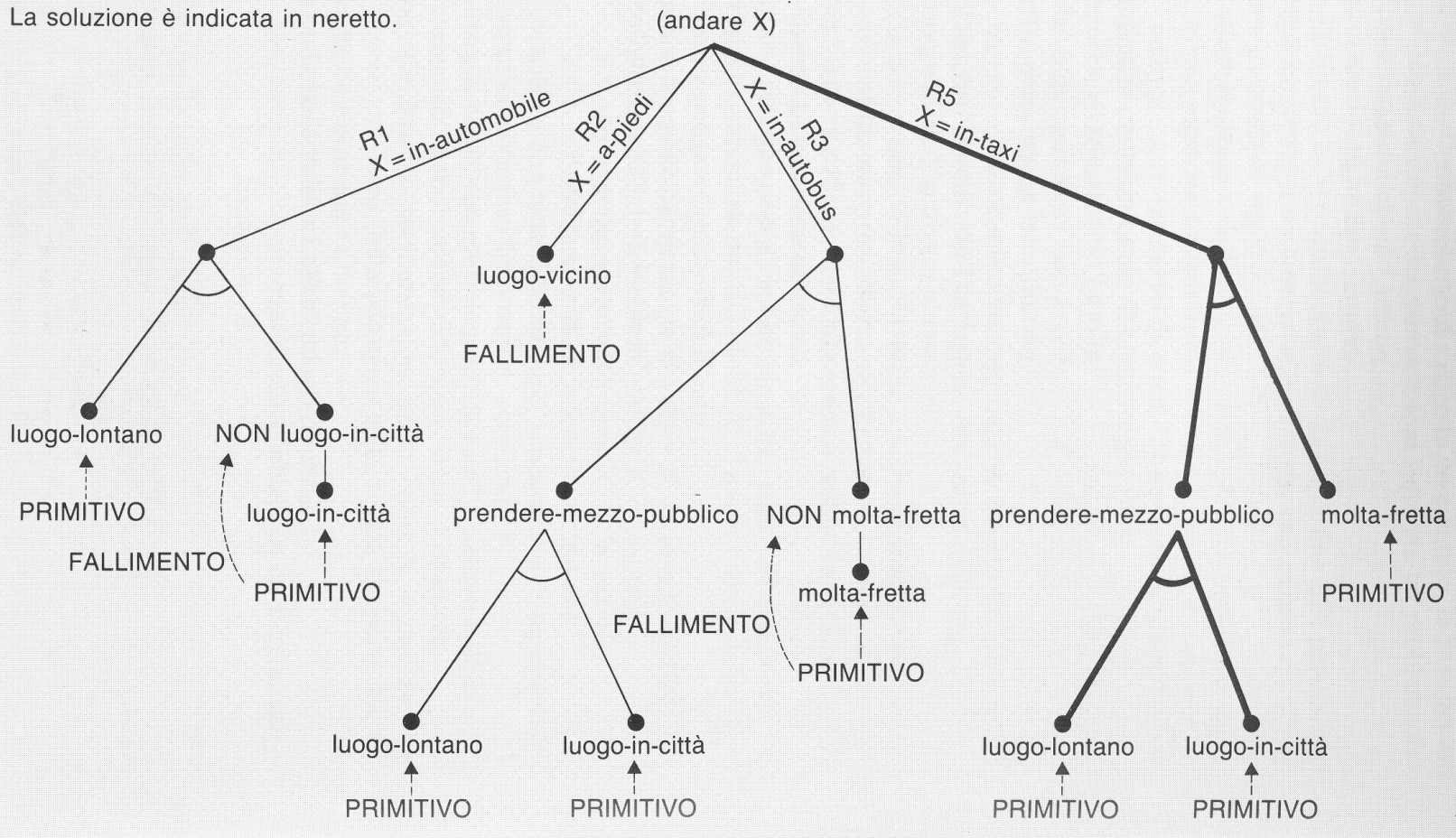
- ✿ Quando si verifica un fallimento, il sistema ritorna sui suoi passi cercando una strada alternativa per risolvere il problema.
- ✿ Il metodo più spesso adottato (backtracking) consiste nel tornare al nodo OR immediatamente superiore al punto in cui si è verificato il fallimento, e scegliere un altro ramo dell'albero (corrispondente a un'altra regola di produzione).
- ✿ Nel nostro caso l'interprete può decidere di riaffrontare il problema iniziale usando la regola R2, ovvero assumendo "**X = a-piedi**".

## I SISTEMI DI PRODUZIONI BACKWARD CHAINING

- ✿ Nel nostro caso l'interprete può decidere di riaffrontare il problema iniziale usando la regola R2, ovvero assumendo "**X = a-piedi**".
- ✿ Si può verificare facilmente che anche questa scelta porta a un fallimento: "**luogo-vicino**" non è nella base dei fatti, né esistono regole che consentano di dedurlo.
- ✿ Analogamente fallisce R4, perché "**molta-fretta**" è presente nella base dei fatti.
- ✿ Le cose vanno diversamente con R5, che porta finalmente alla soluzione del problema: "**X = in-taxi**".

# I SISTEMI DI PRODUZIONI BACKWARD CHAINING

La soluzione è indicata in neretto.





# I SISTEMI DI PRODUZIONI BACKWARD CHAINING

- ✿ Nella sua attività di riduzione di problemi a sotto-problemi l'interprete backward chaining si trova spesso di fronte a scelte alternative, che corrispondono ai nodi OR dell'albero.
- ✿ Qui i criteri di selezione visti nei sistemi forward chaining in genere non possono essere applicati.
- ✿ Infatti, poiché la conclusione di una regola consta di un solo fatto, è difficile introdurre la nozione di specificità di una regola.
- ✿ Inoltre, il procedimento risolutivo non provoca in genere l'inserimento di fatti nuovi nella base dei fatti, e quindi non si può applicare il criterio di scelta dei fatti più recenti.
- ✿ Infine, in molti casi si vuole applicare ripetutamente la stessa regola.

# I SISTEMI DI PRODUZIONI BACKWARD CHAINING

- ✿ Tutto ciò non significa che non si possono trovare dei buoni criteri di scelta della regola da attivare, ma questi dipenderanno spesso dalla particolare natura del problema affrontato.
- ✿ Nel nostro esempio abbiamo adottato un criterio generico che si può applicare spesso, ma che dà luogo a processi risolutivi non molto efficienti: scegliere sempre la prima regola fra quelle identificate, ritornando eventualmente sulla scelta in caso di fallimento.
- ✿ Questo criterio, che corrisponde ad una costruzione in profondità (**depth first**) dell'albero **AND/OR**, è seguito ad esempio dall'interprete del linguaggio Prolog che è essenzialmente un sistema backward chaining.