

Abbiamo bisogno di uno strumento che sia in grado di ragionare sulle azioni, sulle loro applicabilità e i loro effetti.

→ non previsto dalla rappresentazione ontologica.

## Pianificazione (Planning)

- è una branca della IA molto ampia
- il mondo di interesse si trova in una certa situazione in cui valgono determinate proprietà → noi vogliamo modificare questo mondo
- ci limitiamo alla pianificazione delle azioni di un singolo agente

Vogliamo capire quali azioni sono applicabili sulla situazione  $S_1$  per raggiungere la situazione  $S_2$  in cui vale una proprietà obiettivo.

↳ determinare una sequenza di azioni che permette di passare da una situazione corrente ad una situazione in cui vale una certa proprietà obiettivo

- GOAL ACHIEVEMENT (problem solving)
- mantenere vera una condizione

Tipicamente si interagisce col mondo reale ⇒ NO BACKTRACKING

↳ le azioni consumano risorse

Linguaggio di pianificazione :

**PDDL** (Planning Domain Definition Language)

↳ Si ispira al Situation Calculus

# Situation Calculus

- e' stato la prima proposta di formalismo di rappresentazione di contesti di planning e il primo strumento di inferenza
- si basa sulla logica del primo ordine

Prevede 4 elementi cardine:

- **AZIONE** : qualcosa che puo' essere compiuto e influenza il mondo
- **SITUAZIONE** : stato di cose
- **FLUENTE** : predicato il cui valore di verita' puo' cambiare
- **PREDICATO ATEMPORALE** : condizione immutabile

## AZIONE

$MOVE(R, L_1, L_2)$

sposta l'oggetto R dalla pos.  $L_1$  alla pos.  $L_2$



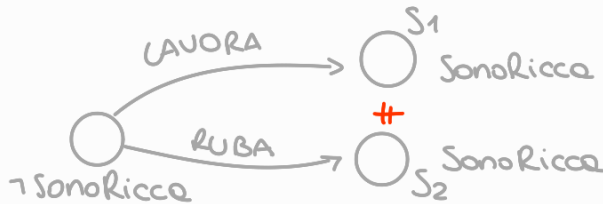
↓  
in FOL e' una FUNZIONE



funzione che fa riferimento ad un oggetto intangibile

**SITUAZIONE** : effetto dell'applicazione di una sequenza di azioni

↓  
 $Do([], s) = s$  → se non faccio nulla il mondo resta così com'è  
 $Do([a|resto], s) = Do([resto], Risultato(a, s))$   
il percorso (il come) e' significativo



> stesso risultato ma due situazioni diverse

Assioma:

$$[Do(Azioni 1, s_1) = Do(Azioni 2, s_2)] \Leftrightarrow (Azioni 1 = Azioni 2) \wedge (s_1 = s_2)$$

AZIONE:

• Assioma di applicabilit 

$$\forall \text{PARAMS}, s \quad \text{APPLICABLE}(\text{ACTION}(\text{PARAMS}), s) \iff \text{PRECOND}(\text{PARAMS}, s)$$

$$\forall x, y, s \quad \text{APPLICABLE}(\text{GO}(x, y), s) \iff \underbrace{\text{AT}(x, s)}_{\text{Fluente}} \wedge \underbrace{\text{ADJACENT}(x, y)}_{\text{pred. atemporale}}$$

• Assioma di effetto

$$\forall \text{PARAMS}, s \quad \text{APPLICABLE}(\text{ACTION}(\text{PARAMS}), s) \Rightarrow \underbrace{\text{EFFECTS}(\text{PARAMS}, \text{RESULT}(\text{ACTION}(\text{PARAMS}), s))}_{\substack{\text{formula} \\ \text{congiunzione di} \\ \text{f. atomiche}}} \quad \downarrow \substack{\text{situazione risultato} \\ \text{dell'applicazione} \\ \text{dell'azione}}$$

$$\forall x, y, s \quad \text{APPLICABLE}(\text{GO}(x, y, s)) \Rightarrow \text{AT}(y, \text{RESULT}(\text{GO}(x, y), s))$$

Esempio sul mondo dei blocchi

$\text{MOVE}(x, y, z)$  : sposta  $x$  da  $y$  a  $z$

Ass. applicabilit 

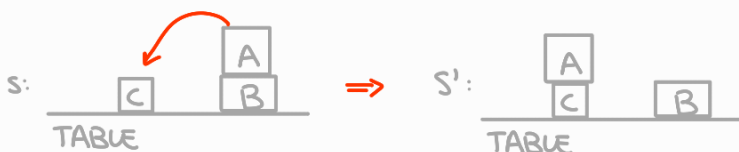
$$\forall x, y, z \quad \text{APPLICABLE}(\text{MOVE}(x, y, z), s) \iff \text{CLEAR}(x, s) \wedge \text{CLEAR}(z, s) \wedge \text{ON}(x, y, z) \wedge x \neq z \wedge y \neq z \wedge x \neq \text{TABLE}$$

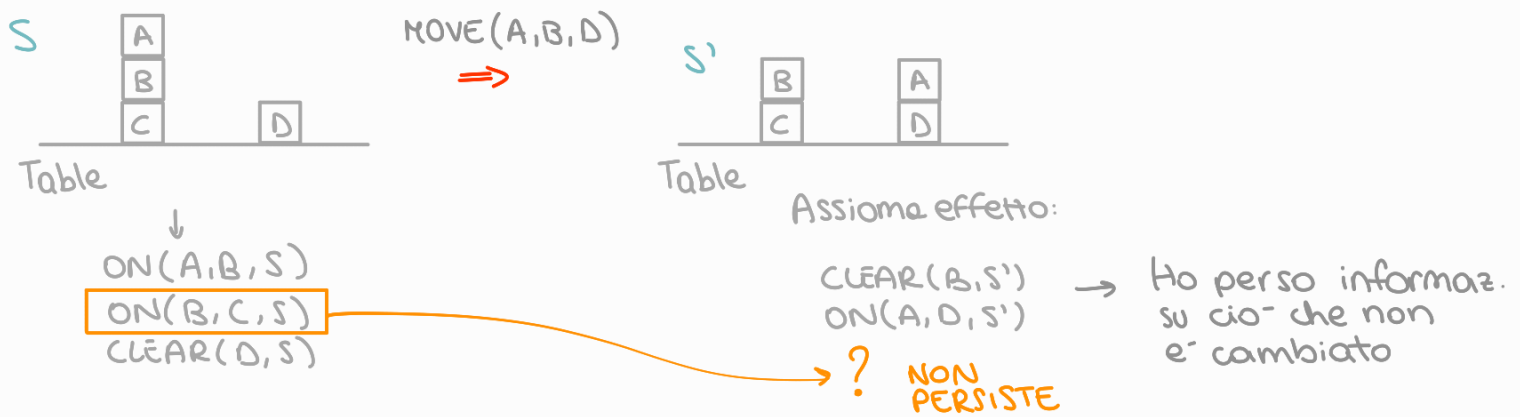


$\text{MOVE}(A, B, C)$  APPLICABLE?  $\checkmark$

Ass. di effetto

$$\forall x, y, z, s \quad \text{APPLICABLE}(\text{MOVE}(x, y, z), s) \Rightarrow \text{ON}(x, z, \text{RESULT}(\text{MOVE}(x, y, z), s)) \wedge \text{CLEAR}(y, \text{RESULT}(\text{MOVE}(x, y, z), s))$$





Ci serve qualche elemento di conoscenza in più per gestire ciò che non cambia → **FRAME PROBLEM**

## • Approccio per enumerazione

azione : impatto limitato → voglio ragionare su ciò che non viene modificato dall'azione

### Assioma di frame

$\forall \text{PARAMS, VARS, S}$

$\text{FLUENT}(\text{VARS}, \text{S}) \wedge \text{PARAMS} \neq \text{VARS} \Rightarrow \text{FLUENT}(\text{VARS}, \text{RESULT}(\text{ACTION}(\text{PARAMS}, \text{S})))$

Se in una situazione vale un certo fluente e in quella situazione viene applicata un'azione su elementi diversi da quelli del fluente, allora il fluente su quelle variabili rimane vero nella situazione risultante.

→ Vale per ogni fluente che non riguarda l'oggetto dell'azione

Poco pratico : 1) Tantissime formule

2) se arricchiamo la KB con fluente nuovo, allora devo aggiungere tanti assiomi di frame

### Assioma di Stato Successore

Se azione applicabile  $\Rightarrow$

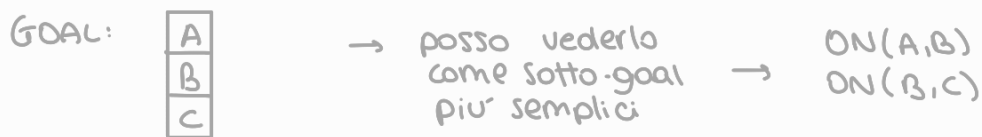
$\left( \text{FLUENTE vero nella situazione risultante} \Leftrightarrow (\text{l'azione lo rende vero}) \vee \begin{matrix} \text{(era vero e l'azione} \\ \text{non l'ha reso falso)} \end{matrix} \right)$

Quando si va a descrivere gli effetti di un'azione, questi si dividono in:

- Aggiunte = cosa viene reso vero
- Cancellazioni = cosa viene reso falso

## CONSEGUIRE GOAL COMPLESSI

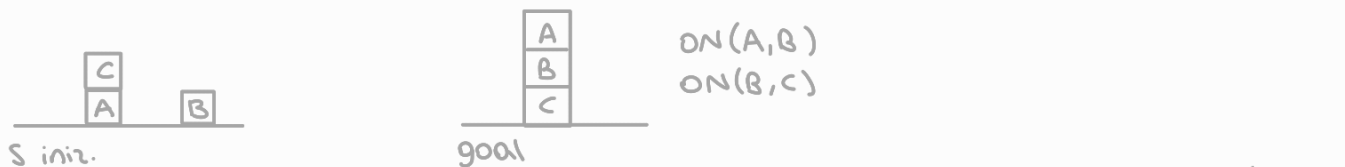
Goal complesso = congiunzione di proprietà



Strategia lineare :

- divido il goal complesso in sottogoal
- trovo dei piani per conseguire ciascuno dei sotto-goal
- li eseguo in sequenza

⚠ NON SEMPRE SUFFICIENTE



1) Supponiamo di eseguire prima  $ON(A,B)$  →



Ora, per eseguire  $ON(B,C)$  devo disfare ciò che ho fatto prima !

2) Supponiamo di eseguire prima  $ON(B,C)$  →



Ancora non va bene, per eseguire  $ON(A,B)$  devo disfare ciò che ho fatto prima



ANOMALIA DI SUSSMAN