Università di Bergamo
Facoltà di Ingegneria

Intelligenza Artificiale

Paolo Salvaneschi

A8_3 V1.3

Rappresentazione della conoscenza

Parte B

Nota: è utilizzato in parte il materiale didattico associato al testo di Stuart J. Russell, Peter Norvig

INDICE

- Rappresentare azioni, situazioni, eventi
- Situation calculus: ontologia
- Esempio
- Problemi del situation calculus
- Applicazioni e limiti

Rappresentare azioni, situazioni, eventi

- Rappresentare il cambiamento, il comportamento dei sistemi
- Ragionamento temporale
- Stati, azioni, risultati delle azioni
- Pianificazione

- Situation Calculus
  - Un modo per descrivere il cambiamento in FOL
  - Un dialetto di FOL

- Situation Calculus
- Formalizzazione in FOL di stati, azioni e effetti di azioni sugli stati
- Utilizzo di un sistema deduttivo per rispondere a domande del tipo:
  - Esiste uno stato che soddisfa date proprietà?
  - Dedurre l'effetto di una data sequenza di azioni (proiezione)
  - Come uno stato presente può essere trasformato in uno stato possibile attraverso delle azioni? (pianificazione)

Situation calculus: ontologia

Esempio: tipo di problema

$s0$: Bob has book1.
John has book2.
Mary has book3.

$e1$: Bob gives book1 to John.

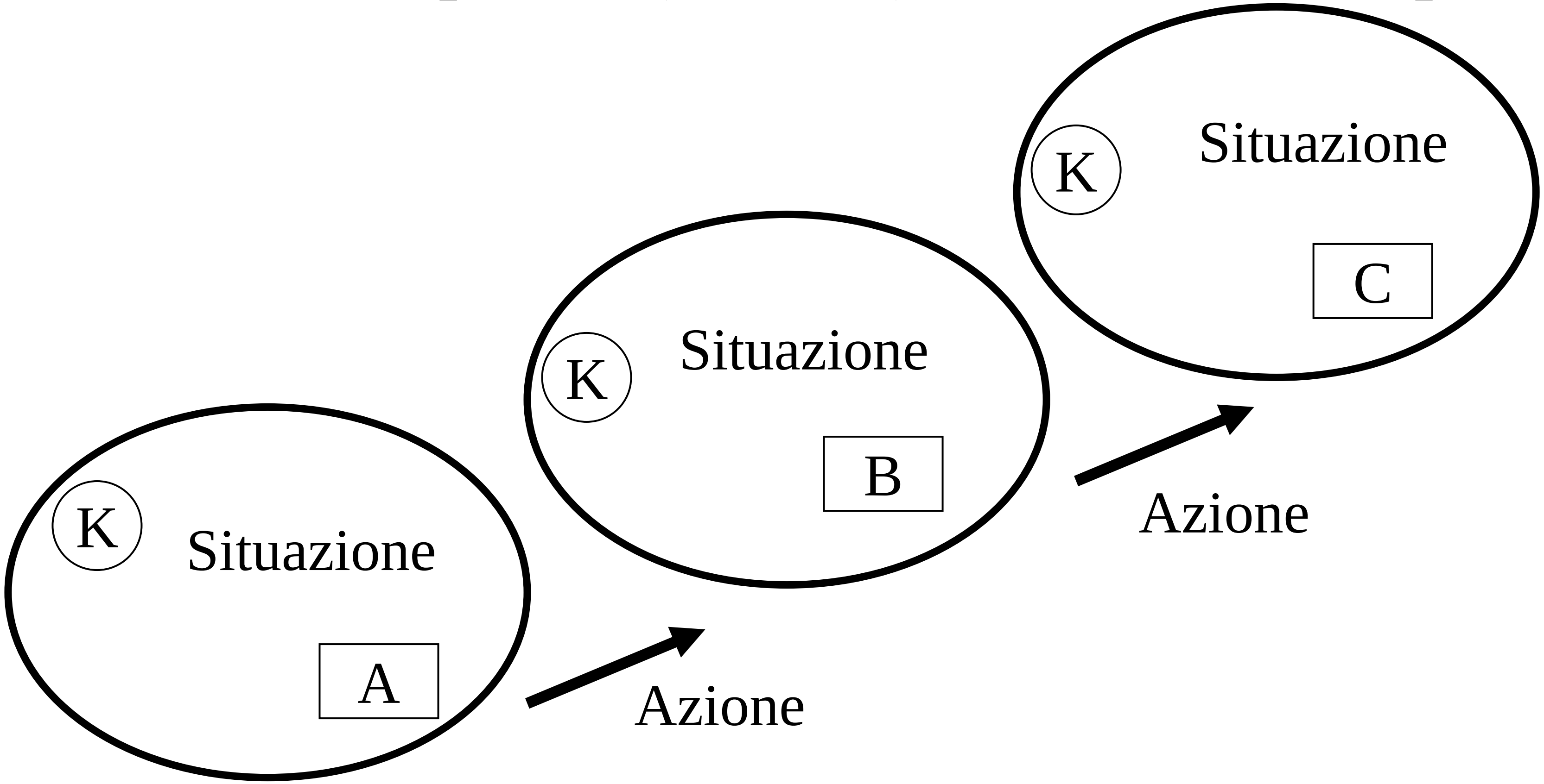
$e2$: John gives book1 to Mary.

s_{end}: Mary has book1. Other properties have remained unchanged.

This results in a sequence of state transitions.

Bob has book1. John has book2. Mary has book3.	$\xrightarrow{e_1}$	**John has book1.** John has book2. Mary has book3.	$\xrightarrow{e_2}$	**Mary has book1.** John has book2. Mary has book3.

- Ontologia: Azioni, Situazioni, Predicati temporali (Fluenti), Predicati atemporali



Situation calculus: ontologia

- Situazioni

  Variabile *s* che rappresenta una situazione

  S0 Situazione iniziale

- Azioni

  Variabile *a* che rappresenta una azione

  La funzione *Result (a, s)* [oppure *Do (a, s)*] denota la situazione che risulta quando si applica l'azione *a* alla situazione *s*

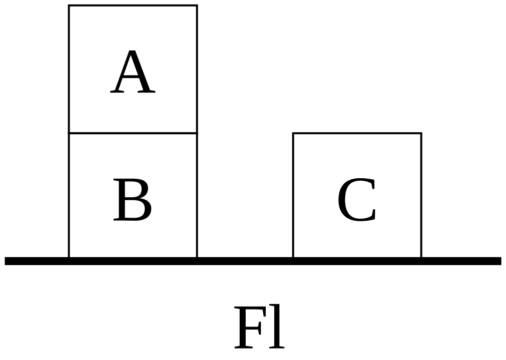
Situation calculus: ontologia

- Fluenti (predicati temporali)

  Funzioni o relazioni che variano da una situazione all'altra

  Su(A, B, S0) *Su(x,y,s)*

  Libero(A, S0) *Libero(x, s)*



- Predicati atemporali

  Non soggetti ad aspetti temporali

  Oro(A)

- Sequenze di azioni

Si può descrivere una sequenza di azioni nei termini del risultato di singole azioni

$$Result\ ([\,],\ s) = s$$

eseguire una sequenza vuota lascia la situazione non modificata

$$Result\ ([a,seq],\ s) = Result\ (seq,\ Result\ (a,\ s)\ )$$

eseguire una sequenza non vuota è eseguire la prima azione e poi le restanti

Situation calculus: ontologia

- Inferenze:
- Proiezione
  - Dedurre il risultato di un definita sequenza di azioni
- Pianificazione
  - Utilizzando un algoritmo di inferenza costruttivo, trovare la sequenza che raggiunge il risultato desiderato

- Ogni azione è descritta da due assiomi:
  - Possibility axiom. Quando è possibile eseguire un’azione?

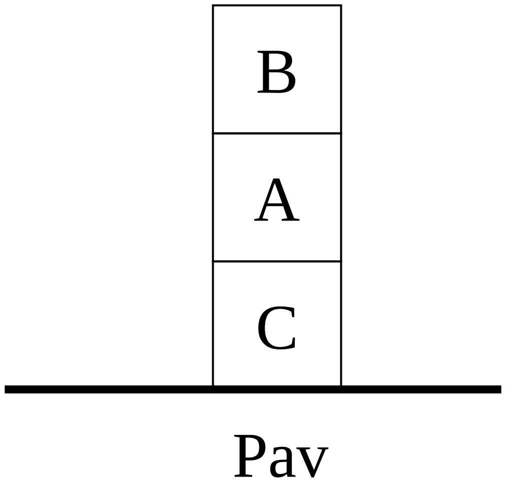
    $$Preconditions \Rightarrow Poss\ (a, s)$$

    Date le precondizioni è possibile eseguire l’azione *a* nella situazione *s*
  - Effect axiom. Quali sono i cambiamenti che conseguono da un’azione?

    $$Poss\ (a, s) \Rightarrow Postconditions$$

    Se una azione è possibile, certe proprietà risulteranno dall’esecuzione dell’azione

Esempio



Descrizione dello stato del mondo

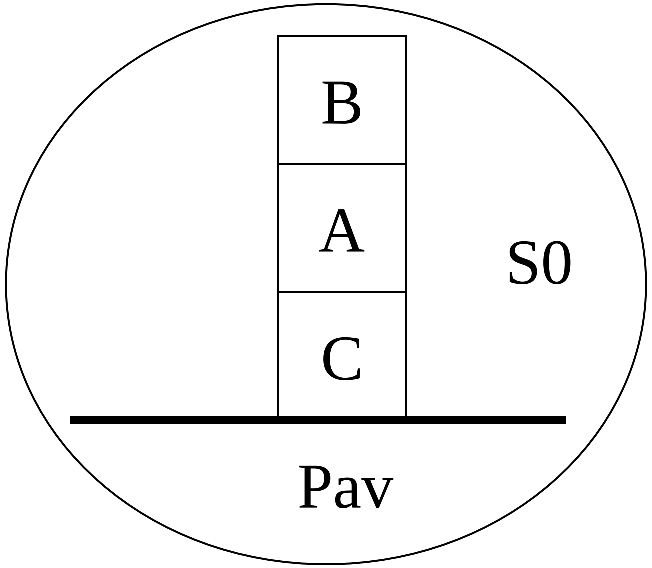
Su(B, A)
Su(A, C)
Su(C, Pav)
Libero(B)
Libero(Pav)

Problema: descrivere il cambiamento

Sposta (B, A, Pav) sposta B da sopra A a sopra Pav

- Vogliamo mantenere la KB monotona: non possiamo rimuovere fatti (es Su(B, A) non più vero) (usare inferenze locali)
- Vorremmo modellare gli stati del mondo prima e dopo le azioni (senza perdere l'informazione degli stati precedenti) (costruire un piano)

Esempio



Lo stato del mondo è un oggetto (reificazione)
Situazione (S0)
Affermazioni vere nella situazione S0:

Su(B, A, S0)
Su(A, C, S0)
Su(C, Pav, S0)
Libero(B, S0)
Libero(Pav, S0)

Affermazioni vere in tutti gli stati. Es:

$(\forall s)$ *Libero (Pav, s)*

Pav è sempre disponibile per posare sopra qualcosa

- S0 Una particolare configurazione del mondo.
- Un punto nel tempo (ontologia del tempo: non durata, stati)

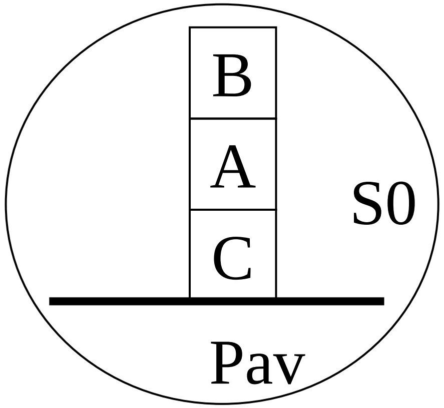
Esempio

- Azioni

Sposta (B, A, Pav) Azione

Sposta(x, y, z)

Famiglia di azioni. Schema con variabili



Result (Sposta(x, y, z), s)

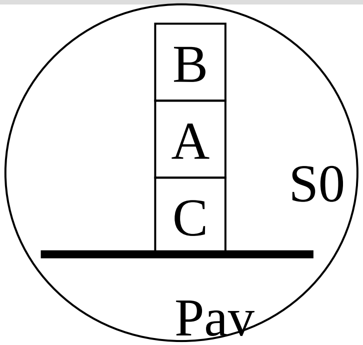
Funzione che fa corrispondere all'azione

Sposta(x, y, z) e alla situazione *s* una nuova situazione

- Fluenti

Su(x, y, s)

Libero(x, s)



- Descrizione delle Azioni
  per ogni coppia azione-fluente

(si assume che tutte le variabili siano quantificate universalmente)

Fluente *Su(x, y, s)* (possibility and effect axioms)

Assioma dell'effetto positivo

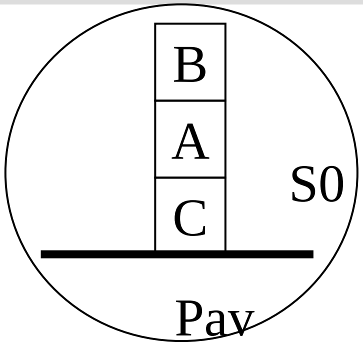
Descrive come l'azione rende vero un fluente

[1]

$Su(x, y, s) \wedge Libero(x, s) \wedge Libero(z, s) \wedge (x \neq z)$
$\Rightarrow Poss\ (Sposta(x, y, z), s)$

$Poss\ (Sposta(x, y, z), s) \Rightarrow Su(x, z, Result\ (Sposta(x, y, z), s)\ )$

Es. il risultato è che B è su Pav nella nuova situazione derivante dall'azione



- Descrizione delle Azioni
  per ogni coppia azione-fluente

(si assume che tutte le variabili siano quantificate universalmente)

Fluente *Su(x, y, s)* (possibility and effect axioms)

Assioma dell'effetto negativo

Descrive come l'azione rende falso un fluente

[2]

$Su(x, y, s) \wedge Libero(x, s) \wedge Libero(z, s) \wedge (x \neq z)$

$\Rightarrow Poss\ (Sposta(x, y, z), s)$

$Poss\ (Sposta(x, y, z), s) \Rightarrow \neg Su(x, y, Result\ (Sposta(x, y, z), s)\ )$

Es. il risultato è che B non è più su A

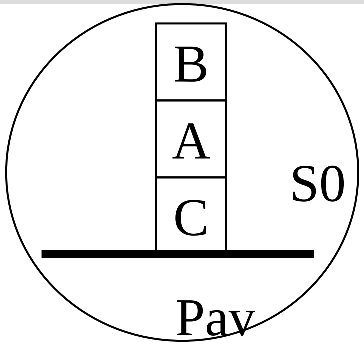
Esempio

Fluente *Libero(x, s)*

Assioma dell'effetto positivo

e dell'effetto negativo

L'effetto non c'è se
prendo il blocco
e lo rimetto al suo posto



[3]

$Su(x, y, s) \wedge Libero(x, s) \wedge Libero(z, s) \wedge (x \neq z) \wedge (y \neq z)$

$\Rightarrow Poss\ (Sposta(x, y, z), s)$

$Poss\ (Sposta(x, y, z), s) \Rightarrow Libero(y, Result\ (Sposta(x, y, z), s)\ )$

Es. il risultato è che A è Libero

[4]

$Su(x, y, s) \wedge Libero(x, s) \wedge Libero(z, s) \wedge (x \neq z) \wedge (z \neq Pav)$

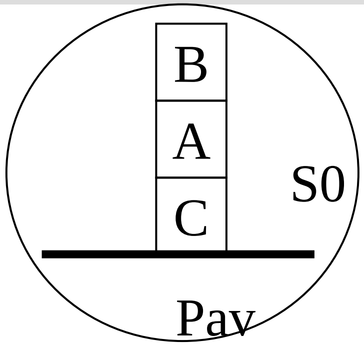
$\Rightarrow Poss\ (Sposta(x, y, z), s)$

$Poss\ (Sposta(x, y, z), s) \Rightarrow \neg Libero(z, Result\ (Sposta(x, y, z), s)\ )$

Es. il risultato è che z non è Libero

(in questo caso Pav è sempre libero-- $(z \neq Pav)$ nessuna azione può renderlo non libero)

Esempio



KB *Gli assiomi precedenti*

fatti
Su(B, A, S0)
Su(A, C, S0)
Su(C, Pav, S0)
Libero(B, S0)
Libero(Pav, S0)

Esempio

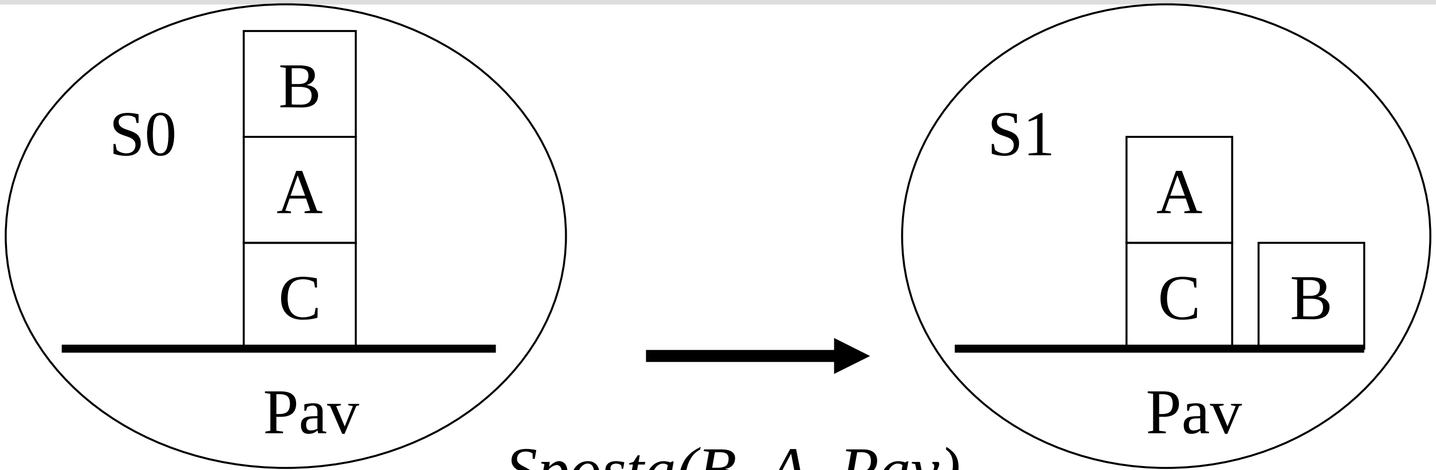
Con le sostituzioni *{ x/B, y/A, z/Pav, s/S0 }*
I fatti soddisfano le precondizioni di [1]

Si inferisce da [1]
Su(B, Pav, Result (Sposta(B, A, Pav), S0))

e si può eseguire l'azione:

Sposta(B, A, Pav)

…………………………..



Sposta(B, A, Pav)

Con le sostituzioni *{ x/B, y/A, z/Pav, s/S0 }*

Si inferisce da [1]

Su(B, Pav, Result (Sposta(B, A, Pav), S0))

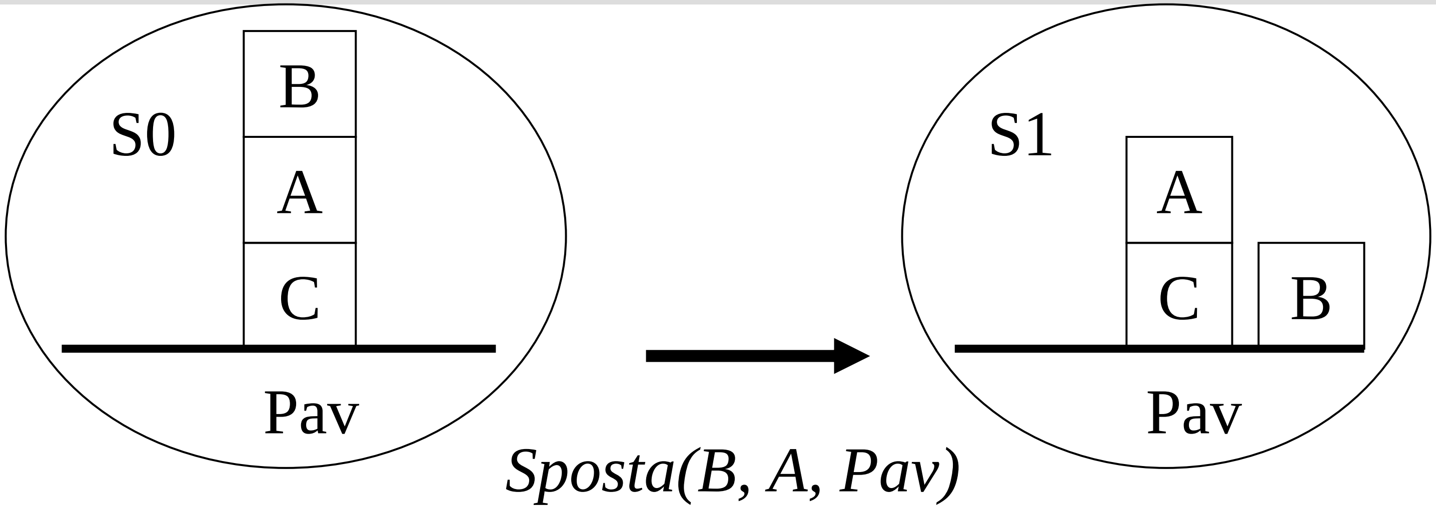
Si inferisce da [2]

$\neg$ *Su(B, A, Result (Sposta(B, A, Pav), S0))*

Si inferisce da [3]

Libero(A, Result (Sposta(B, A, Pav), S0))

Esempio

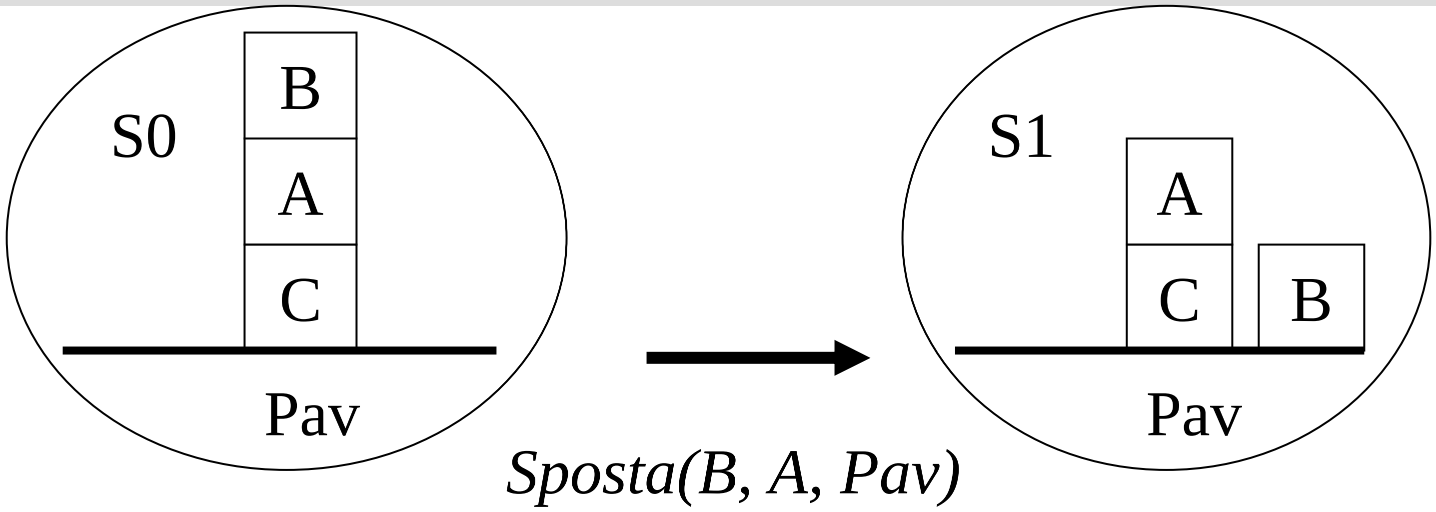


Con le sostituzioni $\{ x/B, y/A, z/Pav, s/S0 \}$

Non si inferisce nulla da [4]

le precondizioni non sono vere: non è vero che $(z \neq Pav)$

Non si inferisce che Pav non è libero; Pav resta libero



Result (Sposta(B, A, Pav), S0) è lo stato dopo l'azione
Denotiamo lo stato con *S1:*
S1= Result(Sposta(B, A, Pav), S0)
Le formule inferite possono essere scritte:

$$Su(B, Pav, S1)$$
$$\neg Su(B, A, S1)$$
$$Libero(A, S1)$$

Esempio

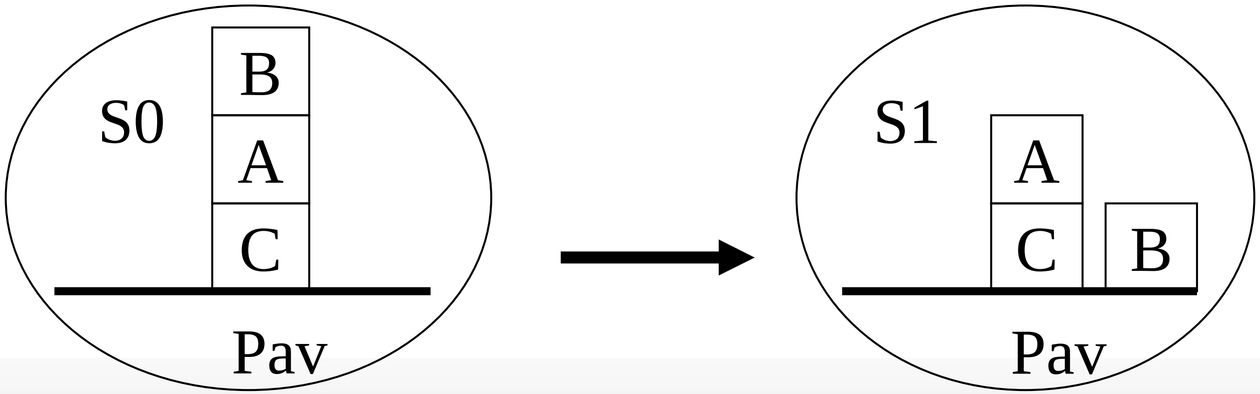
- Nota sulla monotonicità
  - B non è più sopra A, ma tutte le formule scritte prima di eseguire l’inferenza sono ancora vere
  - Ciò che era vero nello stato S0 è rimasto vero
  - Nuovi fatti sono veri nello stato S1

  - KB è monotona

Problemi del situation calculus

- Problemi:
- Frame Problem
  - Come caratterizzare ciò che non cambia a seguito dell'esecuzione di una azione
  - Representational frame problem
  - Inferential frame problem
- Ramification Problem
  - Come caratterizzare quali conseguenze di azioni sopravvivono a successivi cambi di situazioni.
- Qualification Problem
  - Come caratterizzare / quanto dettagliare le condizioni necessarie e sufficienti per una azione
  - Closed world assumption

- Frame Problem (frame-contesto che non cambia rispetto ad un cambiamento)
  - Dopo lo spostamento il fatto che *Su(C, Pav, S0)* era vero nello stato *S0* è ancora vero in *S1*
  - Ma non si può inferire *Su(C, Pav, S1)* , dati gli assiomi scritti sopra
  - Gli assiomi descrivono solo ciò che cambia, non ciò che non cambia



- Prima soluzione:

Scrivere non solo assiomi di effetto, ma anche assiomi di frame (che descrivono ciò che non cambia)

Assiomi di frame per la coppia {azione, fluente}: *{Sposta, Su}*

[5]

$Su(x, y, s) \wedge (x \neq u) \Rightarrow Poss\ (Sposta(u, v, z), s)$

$Poss\ (Sposta(u, v, z), s) \Rightarrow Su(x, y, Result\ (Sposta(u, v, z), s)\ )$

Se un blocco x è su y in S0 lo è anche in S1, se l’azione non lo toglie da y

Es. se C è su Pav in S0 lo è anche in S1

Con questo assioma si può inferire ad es. *Su(A, C, S1)*

Assiomi di frame per la coppia {azione, fluente}: *{Sposta, Su}*

[6]

$\neg Su(x, y, s) \wedge ((x \neq u) \vee (y \neq z)) \Rightarrow Poss\ (Sposta(u, v, z), s)$

$Poss\ (Sposta(u, v, z), s) \Rightarrow \neg Su(x, y, Result\ (Sposta(u, v, z), s))$

Se un blocco x non è su y in S0 non lo è anche in S1, se l'azione sposta un altro blocco oppure non lo pone su y

Es. se A non è su Pav in S0 non lo è anche in S1

Assiomi di frame per la coppia *{Sposta, Libero}*

[7]

$Libero(u, s) \wedge ((u \neq z) \vee (u = Pav)) \Rightarrow Poss\ (Sposta(x, y, z), s)$

$Poss\ (Sposta(x, y, z), s) \Rightarrow Libero(u, Result\ (Sposta(x, y, z), s)\ )$

Se un blocco è Libero in S0 lo è anche in S1, se l'azione non ha posto un blocco su esso (Pav è sempre libero)

[8]

$\neg Libero(u, s) \wedge (u \neq y) \Rightarrow Poss\ (Sposta(x, y, z), s)$

$Poss\ (Sposta(x, y, z), s) \Rightarrow \neg Libero(u, Result\ (Sposta(x, y, z), s)\ )$

Se un blocco non è Libero in S0 lo è anche in S1, se l'azione non ha tolto un blocco da sopra di esso

KB

Assiomi di effetto

[1] $Su(x, y, s) \wedge Libero(x, s) \wedge Libero(z, s) \wedge (x \neq z) \Rightarrow Poss\ (Sposta(x, y, z), s)$
$Poss\ (Sposta(x, y, z), s) \Rightarrow Su(x, z, Result\ (Sposta(x, y, z), s)\ )$

[2] $Su(x, y, s) \wedge Libero(x, s) \wedge Libero(z, s) \wedge (x \neq z) \Rightarrow Poss\ (Sposta(x, y, z), s)$
$Poss\ (Sposta(x, y, z), s) \Rightarrow \neg Su(x, y, Result\ (Sposta(x, y, z), s)\ )$

[3] $Su(x, y, s) \wedge Libero(x, s) \wedge Libero(z, s) \wedge (x \neq z) \wedge (y \neq z) \Rightarrow Poss\ (Sposta(x, y, z), s)$
$Poss\ (Sposta(x, y, z), s) \Rightarrow Libero(y, Result\ (Sposta(x, y, z), s)\ )$

[4] $Su(x, y, s) \wedge Libero(x, s) \wedge Libero(z, s) \wedge (x \neq z) \wedge (z \neq Pav) \Rightarrow Poss\ (Sposta(x, y, z), s)$
$Poss\ (Sposta(x, y, z), s) \Rightarrow \neg Libero(z, Result\ (Sposta(x, y, z), s)\ )$

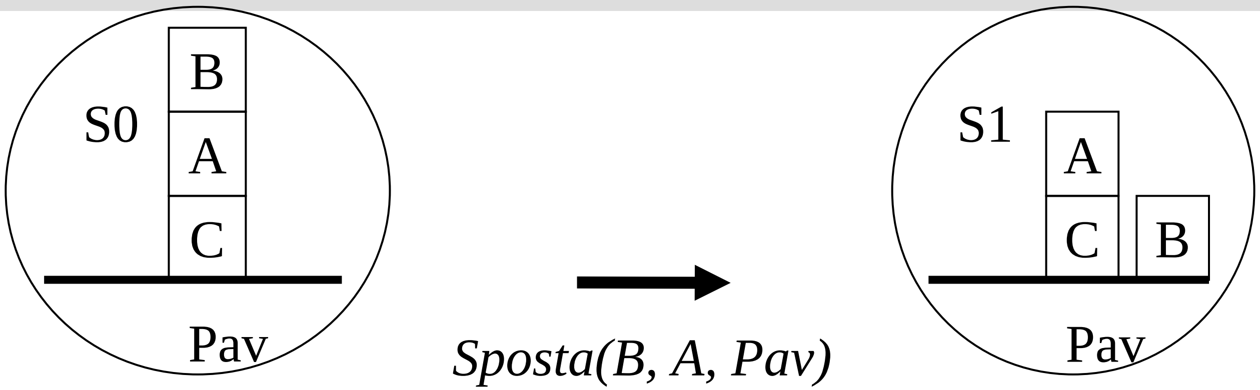
Assiomi di frame

[5] $Su(x, y, s) \wedge (x \neq u) \Rightarrow Poss\ (Sposta(u, v, z), s)$
$Poss\ (Sposta(u, v, z), s) \Rightarrow Su(x, y, Result\ (Sposta(u, v, z), s)\ )$

[6] $\neg Su(x, y, s) \wedge (\ (x \neq u) \vee (y \neq z)\ ) \Rightarrow Poss\ (Sposta(u, v, z), s)$
$Poss\ (Sposta(u, v, z), s) \Rightarrow \neg Su(x, y, Result\ (Sposta(u, v, z), s)\ )$

[7] $Libero(u, s) \wedge ((u \neq z) \vee (u = Pav)) \Rightarrow Poss\ (Sposta(x, y, z), s)$
$Poss\ (Sposta(x, y, z), s) \Rightarrow Libero(u, Result\ (Sposta(x, y, z), s)\ )$

[8] $\neg Libero(u, s) \wedge (u \neq y) \Rightarrow Poss\ (Sposta(x, y, z), s)$
$Poss\ (Sposta(x, y, z), s) \Rightarrow \neg Libero(u, Result\ (Sposta(x, y, z), s)\ )$



KB *Gli assiomi precedenti*

fatti

Su(B, A, S0)
Su(A, C, S0)
Su(C, Pav, S0)
Libero(B, S0)
Libero(Pav, S0)

Denotiamo lo stato con *S1:*
S1= Result(Sposta(B, A, Pav), S0)

Formule inferite:	da
Su(B, Pav, S1)	[1]
¬ *Su(B, A, S1)*	[2]
Libero(A, S1)	[3]
Su(A, C, S1)	[5]
Su(C, Pav, S1)	[5]
Libero(B, S1)	[7]
Libero(Pav, S1)	[7]

esempio

esempio

Su(A, C, S1)

[5]

$Su(x, y, s) \wedge (x \neq u) \Rightarrow Poss\ (Sposta(u, v, z), s)$

$Poss\ (Sposta(u, v, z), s) \Rightarrow Su(x, y, Result\ (Sposta(u, v, z), s)\ )$

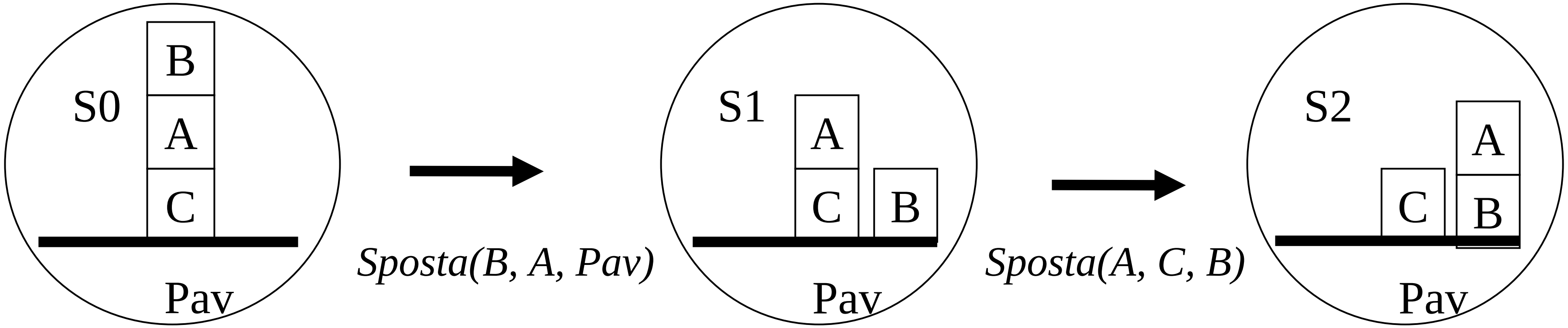
Se un blocco x è su y in S0 lo è anche in S1, se l'azione non lo toglie da y

Con le sostituzioni $\{ x/A, y/C, u/B, v/A, z/Pav, s/S0 \}$

$Su(A, C, S0) \wedge (A \neq B) \Rightarrow Poss\ (Sposta(B, A, Pav), S0)$

$Poss\ (Sposta(B, A, Pav), s) \Rightarrow Su(A, C, Result\ (Sposta(B, A, Pav), S0)\ )$

Esempio



KB	*Gli assiomi precedenti*	
nuovi fatti	*Su(B, Pav, S1)*	*Su(A, C, S1)*
	Libero(A, S1)	*Su(C, Pav, S1)*
	¬ Su(B, A, S1)	*Libero(B, S1)*
		Libero(Pav, S1)

Piano delle azioni per raggiungere l'obiettivo B sopra A

Si può inferire da [1]:

$$Su(A,B,\underbrace{Result(\underline{Sposta(A, C, B)},\underbrace{Result\ (\underline{Sposta(B, A, Pav)}, S0)}_{S1})}_{S2})$$

Problemi del situation calculus

- Problema
  - Una coppia di assiomi per ogni combinazione di fluente e azione – $O(AF)$ F fluenti, A azioni
  - Gli assiomi di frame servono per inferire che una proprietà di una situazione resta vera se una azione che cambia la situazione non ha effetto su quella proprietà
  - In una grande KB con cambiamenti locali (situazione comune per problemi reali): grande quantità di assiomi di frame – inefficienza

Problemi del situation calculus

- Ogni azione *A* ha al massimo *E* effetti
- *E* è tipicamente molto minore di *F*
- Bisognerebbe saper rappresentare ciò che accade con una KB più piccola di dimensione *O(AE)* e non *O(AF)*

Problemi del situation calculus

- Ridurre la quantità di assiomi di frame
  - Si specifica il valore di verità di ogni fluente nello stato successivo in funzione dell'azione e del valore di verità nello stato corrente
  - Non si specifica l'effetto di ogni azione, ma come ogni fluente evolve nel tempo
  - Successor-state axioms – Dimensione di KB: *O(AE)* letterali, *E* effetti, *A* azioni

L'azione è possibile $\Rightarrow$
(il fluente è vero nello stato risultante
$\Leftrightarrow$ l'effetto dell'azione lo ha reso vero $\vee$
era già vero e l'azione non lo ha modificato)

Se e solo se

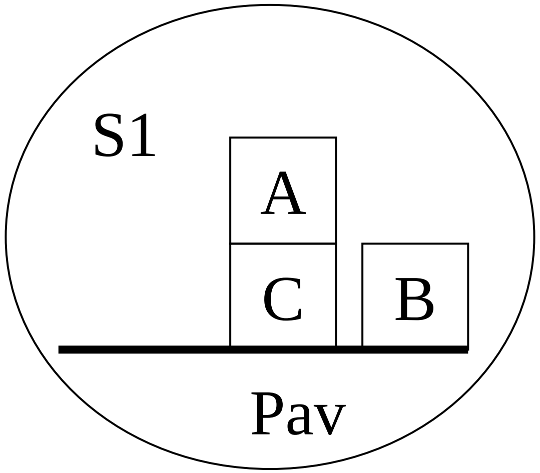
x è su *y* come risultato dell'azione *a* se e solo se

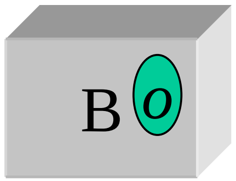
ho spostato *x* su *y*

$$Poss(a, s)\ ) \Rightarrow$$
$$(\ Su(x, y, Result(a, s)) \Leftrightarrow a=(Sposta(x, y, z) \vee$$
$$(Su(x, y, s) \wedge a \neq Sposta(x, y, z)))$$

x era già su *y* e non l'ho spostato

- Ramification Problem
  - Come caratterizzare quali conseguenze di azioni sopravvivono a successivi cambi di situazioni.





B contiene un oggetto *o*

Spostare B ha come effetto implicito spostare *o*

o si trova in z se si trovava già in z oppure se è stato trasportato da B….

Problemi del situation calculus

- Ridurre la quantità di assiomi di frame:
  - Representational frame problem: Assiomi di frame
- Problema di trattare computazionalmente i frame axioms
  - Inferential frame problem
  - Dato un piano p di t passi tale che $S_t = Result(p, S_0)$ bisogna considerare ognuno degli F assiomi in ogni istante temporale t

    Inefficienza: in gran parte si copiano fluenti immutati da una situazione all'altra

……………….

Problemi del situation calculus

- Qualification Problem
  - Come caratterizzare / quanto dettagliare le condizioni necessarie e sufficienti per una azione
  - Negli assiomi di effetto sono state specificate tutte le condizioni necessarie per il successo di un'azione?

> Andare in Milano:
> Avere il biglietto del treno
> Andare alla stazione in orario
> …..
> Nessun incidente sulla strada
> …..

  - Closed world assumption

- Closed world assumption
- Le formule atomiche di cui non si afferma la verità sono considerate false

Andare in Milano:

Vere → Avere il biglietto del treno

Andare alla stazione in orario

…..

False → Nessun incidente sulla strada

…..

- Ciò è vero per i DataBase, non per FOL
  - O si dice esplicitamente (completamento di Clark)
  - O è assunto dall'implementazione del motore di inferenza

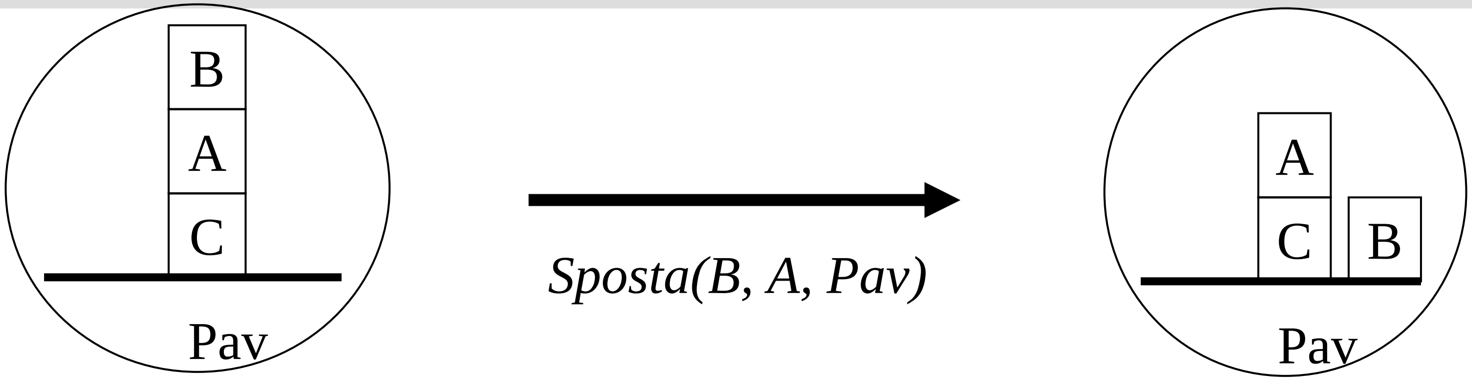
Applicazioni e limiti

- Generazione di piani
- Utilizzo del situation calculus (in linea di principio):

  Dato uno stato obiettivo, dimostriamo (refutazione con risoluzione) che esiste lo stato obiettivo ed usiamo il predicato di risposta per estrarre lo stato nei termini di una funzione delle azioni che lo producono
- Utilizzato nei primi sistemi di pianificazione Non più usato. Importante per esporre i concetti
- Sistemi di pianificazione: fusione dell’approccio del situation calculus e della ricerca nello spazio degli stati: STRIPS

Applicazioni e limiti

- Idea di STRIPS (Fikes, Hart, Nilsson 1972)
- Un insieme di formule in FOL che descrivono stati del mondo sono considerate uno “stato” (una struttura di dati che può essere modificata da una azione)
- Una azione è descritta da un “operatore STRIPS” costituito da
  - Precondizioni dell’operatore
  - Lista di cancellazioni
  - Lista di modifiche
  - Non modificate
- La costruzione di un piano è la ricerca nello spazio degli stati (ad es in avanti con A*) per trovare uno stato obiettivo



Precondizioni
Su(B, A)
Libero(B)
Libero(Pav)

Se le precondizioni dell'operatore appaiono nello stato, l'operatore può essere eseguito

Lista delle cancellazioni

Su(B, A)
Su(A, C)
Su(C, Pav)
Libero(B)
Libero(Pav)

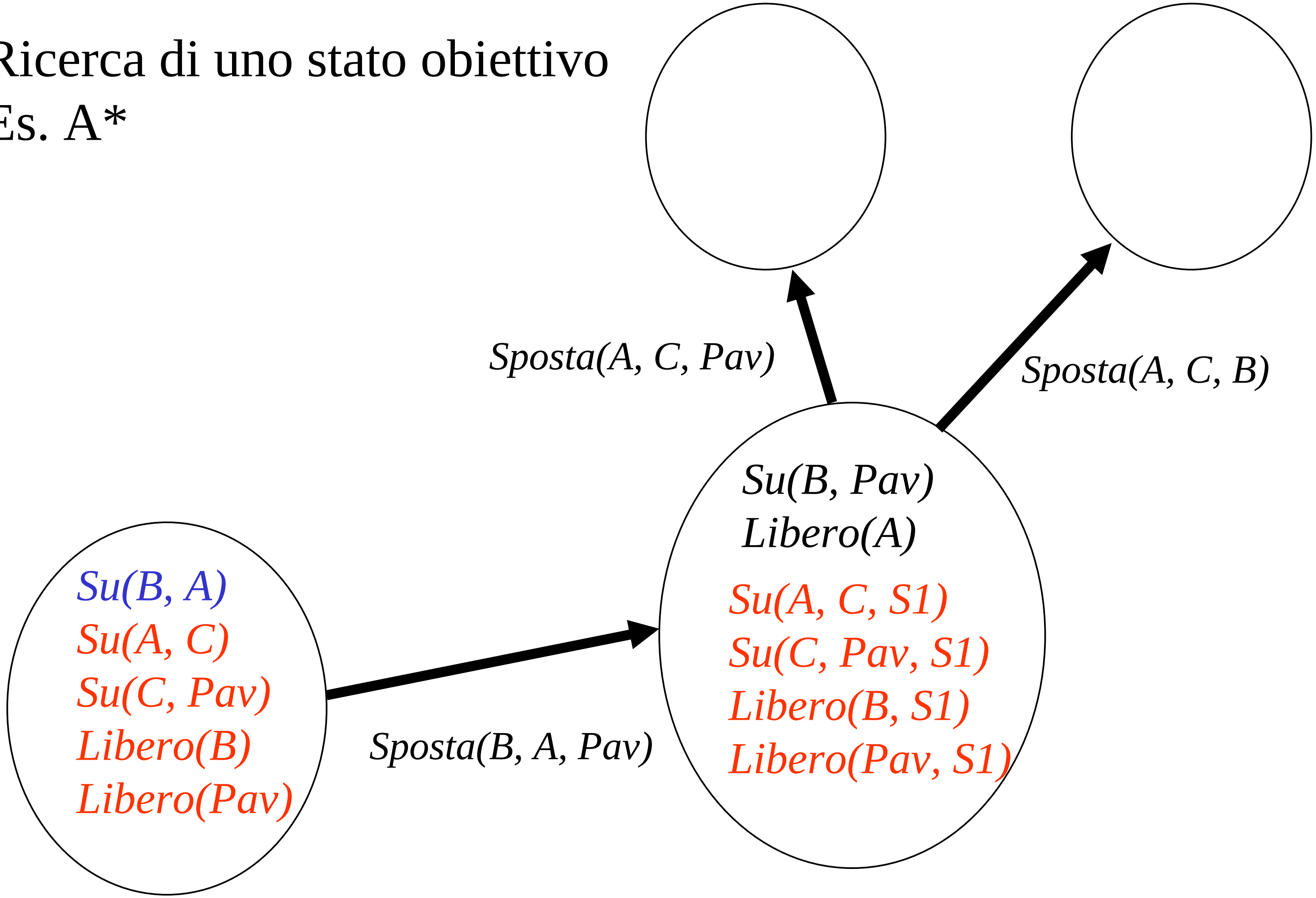
Lista delle aggiunte

Non modificate

Come è affrontato il problema del frame

Su(B, Pav)
Libero(A)
Su(A, C)
Su(C, Pav)
Libero(B)
Libero(Pav)

Ricerca di uno stato obiettivo
Es. A*



Applicazioni e limiti

<u>Un treno sposta dei vagoni tra due città</u>

1. Il treno parte dalla città x
2. Il viaggio dura tipicamente dalle 4 alle 6 ore
3. I vagoni devono restare connessi al treno durante l'intero viaggio
4. Il treno percorre il segmento A1 di linea ferroviarie, attraversa lo scambio J1 e prosegue sul segmento A2 per il resto del viaggio
5. Il treno arriva alla città y

Il calcolo delle situazioni può rappresentare solo 1 e 5
(singolo agente, azioni atomiche e discrete)
2 richiede informazioni di durata
3 richiede precondizioni che restano attive oltre l'attivazione dell'azione
4 richiede postcondizioni che non iniziano alla fine dell'azione
Eventi esterni (interrupts) ?
Concorrenza ?

- Ragionamento temporale e spaziale