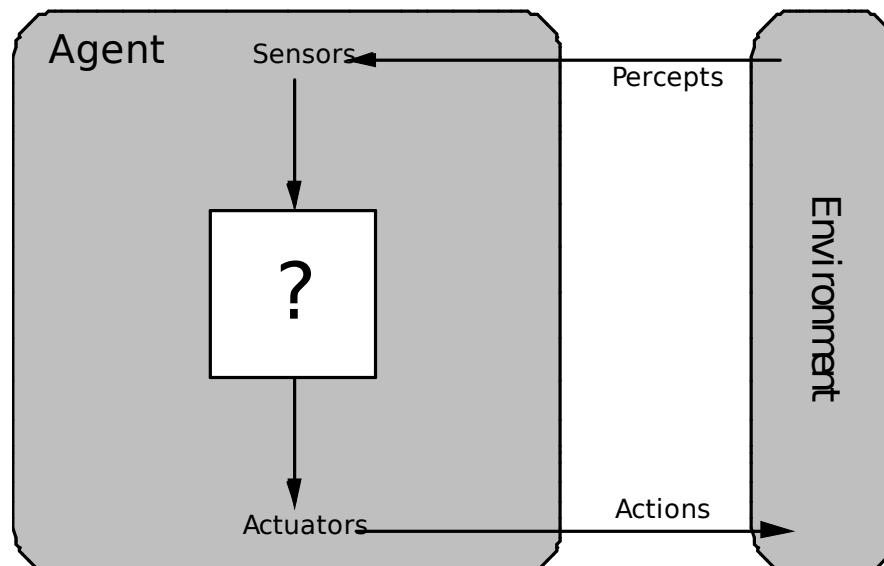


Agenti

si introduce la nozione di agente (intelligente) che opera in un ambiente e si fa vedere come l'agente possa avere sia comportamenti reattivi che deliberativi a seconda del compito assegnato. Si illustra come agente debba avere capacità di risoluzione automatica di problemi e di ragionamento sullo stato del mondo e sul suo stato.

Agente & Ambiente

- **Agente**: astrazione che rappresenta un qualsiasi sistema che percepisce il proprio **ambiente** tramite dei **sensori** ed agisce su di esso tramite degli **attuatori**



Non esistono agenti che non siano situati in un ambiente.

Agente e ambiente costituiscono un binomio inscindibile

Il '?' rappresenta la funzione deliberativa esercitata dall'agente, che permette di determinare l'azione da eseguire

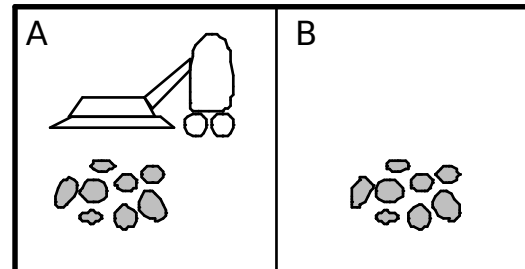
Tipologie ed esempi di sensori fisici

- sensori di luce (o sensori ottici)
- sensori a infrarossi
- sensori di suono: microfoni
- sensori di accelerazione: accelerometri
- sensori di temperatura
- sensori di calore
- sensori di radiazione: contatori Geiger
- sensori di resistenza elettrica: ohmmetri
- sensori di corrente elettrica: galvanometri, amperometri
- sensori di tensione elettricae sensori di potenza
- sensori di magnetismo
- sensori di pressione
- sensori di gas e flusso di liquidi
- sensori di movimento
- sensori di orientamento
- sensori di forza: celle di carico, estensimetri.
- sensori di prossimità
- sensori di distanza
- sensori biometrici
- sensori chimici

Sequenza percettiva

- **Sequenza percettiva**: storia completa delle percezioni di un agente
- Un agente sceglie la prossima azione sulla base della situazione in cui si trova, cioè di quanto ha percepito fino a quel momento
- *Come realizzare il modulo deliberativo?*
- *Tabella percepito - azione:*
talvolta percepito sarà una sequenza percettiva, talvolta sarà possibile prescindere dalla storia e focalizzarsi sulla percezione corrente

Esempio: mondo dell'aspirapolvere

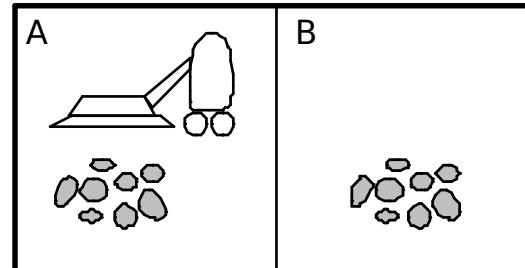


Sequenza percettiva

Azione

[A, pulito]	destra
[A, sporco]	aspira
[B, pulito]	sinistra
[B, sporco]	aspira
[A, pulito][A, pulito]	destra
[A, pulito][A, sporco]	aspira
... seq. di lungh. crescente	...

Altri aspirapolveri?



Sequenza percettiva

Azione

[A, pulito]	aspira
[A, sporco]	aspira
[B, pulito]	aspira
[B, sporco]	aspira
[A, pulito][A, pulito]	destra
[A, pulito][A, sporco]	destra
... seq. di lungh. crescente	...

Se cambio azione produco
molti moduli di deliberazione
differenti, molti agenti diversi

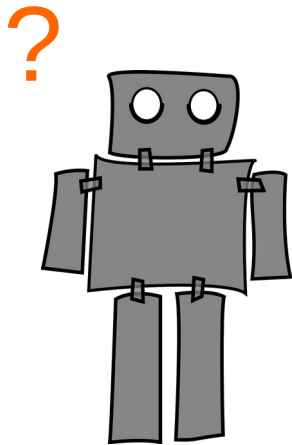
SONO CONFRONTABILI? SULLA BASE DI COSA?

Fare la cosa giusta: cosa vuol dire?

- In termini informali un agente razionale “fa la cosa giusta” — cioè opera per conseguire il “successo”
- Occorre una misura di prestazione (**performance measure**)
- Idea: *misurare la bontà degli stati attraversati*
- La misura della bontà è definita in modo dipendente dall’effetto che si desidera conseguire
- Esempio:

STATO	BONTÀ
[A, pulito], [B, pulito]	1
[A, pulito], [B, sporco]	0.5
[A, sporco], [B, pulito]	0.5
[A, sporco], [B, sporco]	0

Razionalità (comportamento razionale)



Il **comportamento razionale** di un agente dipende da 4 fattori:

1. Azioni nelle facoltà dell'agente
2. Performance measure
3. Conoscenza dell'ambiente
4. Sequenza percettiva

Un **agente razionale** dovrebbe scegliere sempre un'azione che massimizza la misura di prestazione attesa, data la particolare sequenza percettiva in oggetto e le informazioni derivabili dalla conoscenza dell'ambiente

- Quale aspirapolvere è più razionale?
- Sono definibili comportamenti ancora più razionali?
Rispetto a quali effetti desiderati?

Razionalità versus onniscienza/chiaroveggenza

RAZIONALITÀ

Ottimizza il risultato atteso

Possono intercorrere fattori ignoti o imprevedibili che impediscono di conseguire il risultato atteso

Voglio che al compleanno di mio figlio vi sia la torta più bella e meno costosa. Conosco le offerte dei pasticceri della zona e i gusti di mio figlio. Ho sufficiente denaro in tasca. So fare acquisti.

Cosa non posso prevedere:
la nonna porterà una torta identica (regalandola)

Cosa non so
Offerte di una nuova pasticceria

ONNISCENZA/CHIAROVEGGENZA

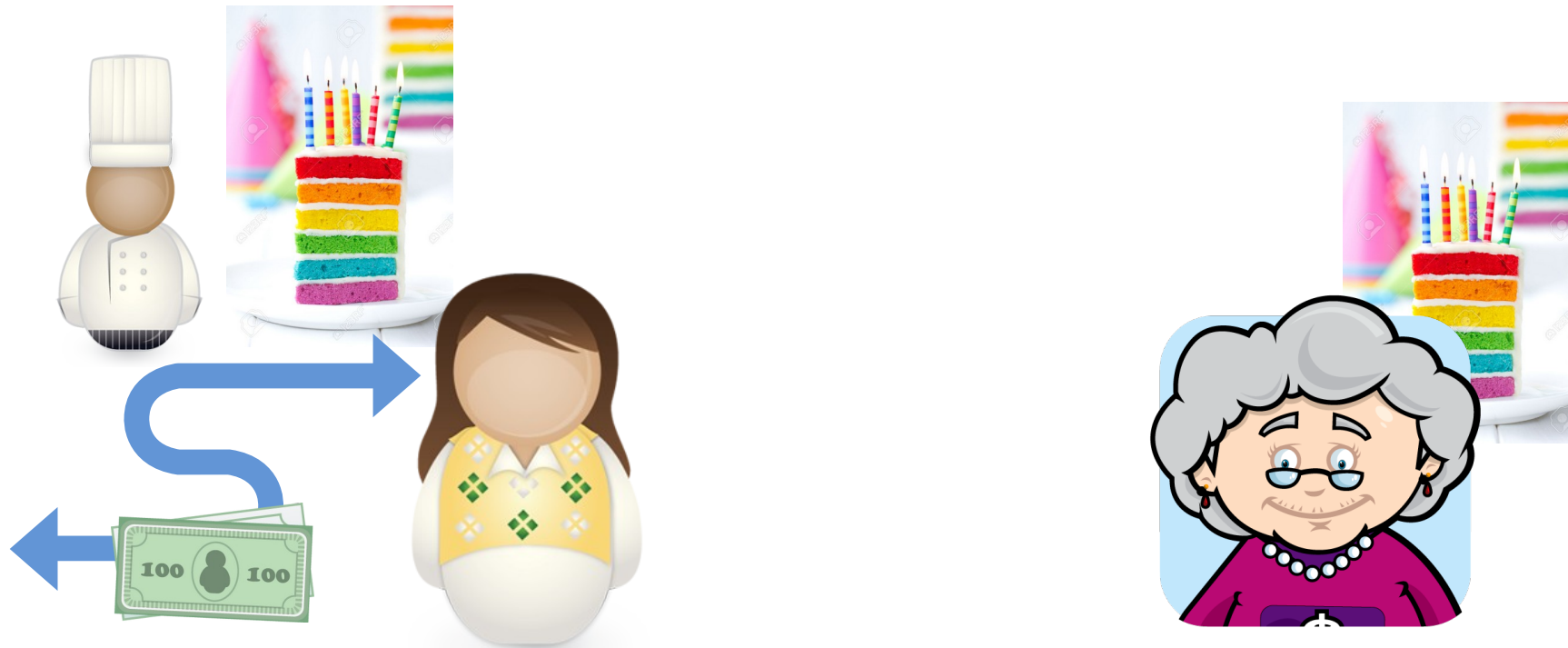
Ottimizza il risultato reale

Non possono intercorrere fattori ignoti o imprevedibili

Es. non compero nessuna torta perché so che la nonna regalerà la torta perfetta. Ottengo il prezzo migliore perché so di una nuova pasticceria che fa delle offerte.

Razionalità non è chiaroveggenza

PASTICCERIA



La mia sequenza percettiva non include il fatto che la nonna faccia una torta quindi il mio comportamento è razionale, anche se avrei potuto raggiungere il risultato in modo più economico non comperando alcuna torta

Razionalità è modificare il comportamento sulla base dell'esperienza

PASTICCERIA



Sapendo che la nonna sa fare la torta perfetta e che potrebbe pensare di farla per il compleanno del nipote, non sarebbe più un comportamento razionale acquistare la torta senza prima verificare se la nonna ne farà una.

Un agente razionale è più efficace se ha la capacità di imparare

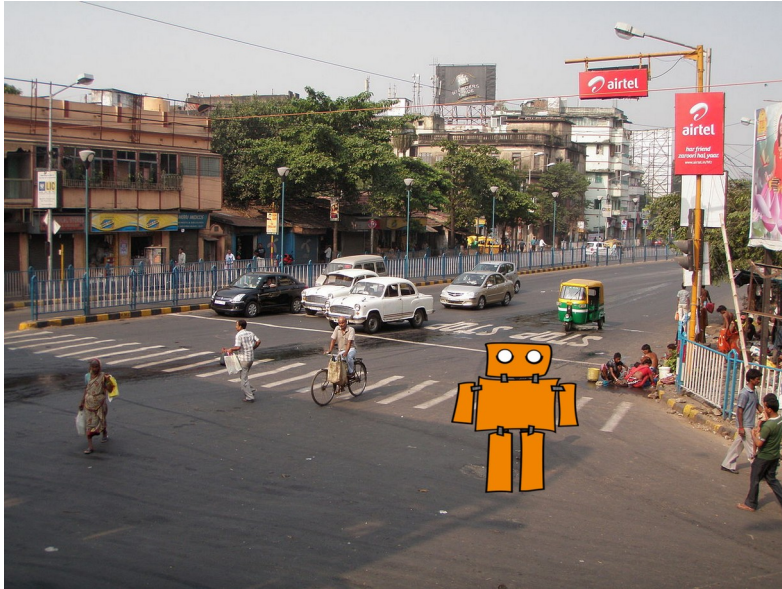
Razionalità non è onniscienza

PASTICCERIA



La mia sequenza percettiva non include il fatto che una nuova pasticceria in un quartiere vicino produca la stessa torta a un prezzo inferiore a quello da me pagato, quindi il mio comportamento è razionale, anche se avrei potuto raggiungere il risultato spendendo meno. **NB: razionale \neq “di successo”**

Attraversare la strada



È razionale se il robot attraversa la strada senza guardare prima a destra?

NO!

Percepire non è un atto passivo

Prima di eseguire un'azione un agente deve decidere se ha abbastanza informazioni o se è necessario eseguire altri atti percettivi

La razionalità dipende dalla sequenza percettiva e può influenzare la sequenza percettiva

Caratterizzazione dell'ambiente

LEGGERE

- **Task environment**: è il contesto in cui l'agente è inserito. Può essere fisico (reale o simulato, es. per agenti robotici) oppure, per esempio, sociale e comprendere le relazioni in essere con altri agenti (es. nel caso di agenti software)
- **PEAS**: *performance, environment, actuators, sensors*. Sono gli elementi che definiscono il task environment, esempio:

Agent Type	Performance Measure	Environment	Actuators	Sensors
Taxi driver	Safe, fast, legal, comfortable trip, maximize profits	Roads, other traffic, pedestrians, customers	Steering, accelerator, brake, signal, horn, display	Cameras, sonar, speedometer, GPS, odometer, accelerometer, engine sensors, keyboard

Figure 2.4 PEAS description of the task environment for an automated taxi.

Altri esempi di PEAS

LEGGERE

Agent Type	Performance Measure	Environment	Actuators	Sensors
Medical diagnosis system	Healthy patient, minimize costs, lawsuits	Patient, hospital, staff	Display questions, tests, diagnoses, treatments, referrals	Keyboard entry of symptoms, findings, patient's answers
Satellite image analysis system	Correct image categorization	Downlink from orbiting satellite	Display categorization of scene	Color pixel arrays
Part-picking robot	Percentage of parts in correct bins	Conveyor belt with parts; bins	Jointed arm and hand	Camera, joint angle sensors
Refinery controller	Maximize purity, yield, safety	Refinery, operators	Valves, pumps, heaters, displays	Temperature, pressure, chemical sensors
Interactive English tutor	Maximize student's score on test	Set of students, testing agency	Display exercises, suggestions, corrections	Keyboard entry

Figure 2.5 Examples of agent types and their PEAS descriptions.

Caratteristiche dell'ambiente

LEGGERE

Completamente osservabile

in ogni istante i sensori danno accesso a tutti gli aspetti dell'ambiente rilevanti per la scelta dell'azione

Parzialmente osservabile

i sensori danno accesso solo a parte dell'informazione rilevante (cause: sensori imprecisi oppure non in grado di rilevare alcuni dati)

Deterministico

lo stato successivo è determinato dallo stato corrente e dall'azione applicata

Stocastico

applicando più volte una stessa azione in uno stesso stato si possono raggiungere stati diversi. Si dice **strategico** quando è stocastico solo per quanto riguarda le azioni degli altri agenti.

Episodico

l'esperienza degli agenti è divisa in episodi atomici: un episodio è dato da una percezione seguita da una singola azione (esempio: classificazione)

Sequenziale

Attività composta da più passi ognuno dei quali in generale influenzerà i successivi

Caratteristiche dell'ambiente

LEGGERE

Statico

l'ambiente non cambia mentre l'agente "pensa" (cioè mentre decide quale azione eseguire)

Dinamico

l'ambiente può cambiare mentre l'agente "pensa"

Discreto

possono essere discreti stato, tempo, percezioni, azioni (esempio: gli scacchi hanno stati, percezioni, azioni discreti)

Continuo

possono essere continui stato, tempo, percezioni, azioni (esempio: gli scacchi hanno tempo continuo)

Singolo Agente

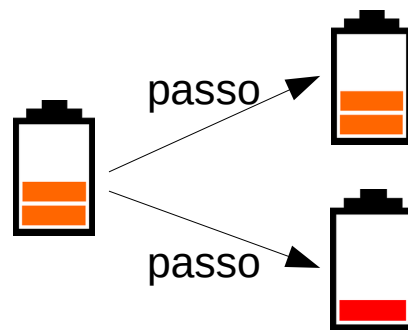
Viene modellata come agente una sola entità

Multiagente

Vengono modellate come agenti più entità

- **Parzialmente osservabile / stocastico:**

vi è una connessione. Spesso viene visto come stocastico un ambiente che è parzialmente osservabile perché non si ha la percezione di quegli aspetti che renderebbero deterministico il mondo. Esempio: batteria con carica bassa, media alta, si consuma in modo continuo ma l'agente percepisce solo i tre stati citati. La stessa azione passo eseguita con carica media potrebbe portare sia a un carica bassa sia a rimanere nello stato carica media.



- **Singolo agente / multiagente:**

talvolta il programmatore deve decidere se modellare alcune entità del mondo come agenti o come elementi dell'ambiente. Quali devono essere rappresentate come agenti? Quelle il cui comportamento tenta di massimizzare una misura di prestazione che dipende anche dal comportamento di altri agenti.

- **Caso più complesso:**

ambiente parzialmente osservabile, stocastico, sequenziale, dinamico, continuo e multiagente.

LEGGERE

Esempio di ambiente: parole crociate

LEGGERE

- Osservabilità:
- Agenti:
- Deterministico:
- Episodico:
- Statico:
- Discreto:

Esempio di ambiente: parole crociate

LEGGERE

- Osservabilità: totale
- Agenti: singolo
- Deterministico: deterministico
- Episodico: sequenziale
- Statico: statico
- Discreto: discreto

Esempio di ambiente: guidare un taxi

LEGGERE

- Osservabilità:
- Agenti:
- Deterministico:
- Episodico:
- Statico:
- Discreto:

Esempio di ambiente: guidare un taxi

LEGGERE

- Osservabilità: parziale
- Agenti: multiagente
- Deterministico: stocastico
- Episodico: sequenziale
- Statico: dinamico
- Discreto: continuo

Esempio di ambiente: tutor di inglese interattivo

LEGGERE

- Osservabilità:
- Agenti:
- Deterministico:
- Episodico:
- Statico:
- Discreto:

Esempio di ambiente: tutor di inglese interattivo

LEGGERE

- Osservabilità: parziale
- Agenti: multiagente
- Deterministico: stocastico
- Episodico: sequenziale
- Statico: dinamico
- Discreto: discreto

Esempio di ambiente: analizzatore di immagini

LEGGERE

- Osservabilità:
- Agenti:
- Deterministico:
- Episodico:
- Statico:
- Discreto:

Esempio di ambiente: analizzatore di immagini

LEGGERE

- Osservabilità: totale
- Agenti: singolo
- Deterministico: deterministico
- Episodico: episodico
- Statico: statico
- Discreto: continuo

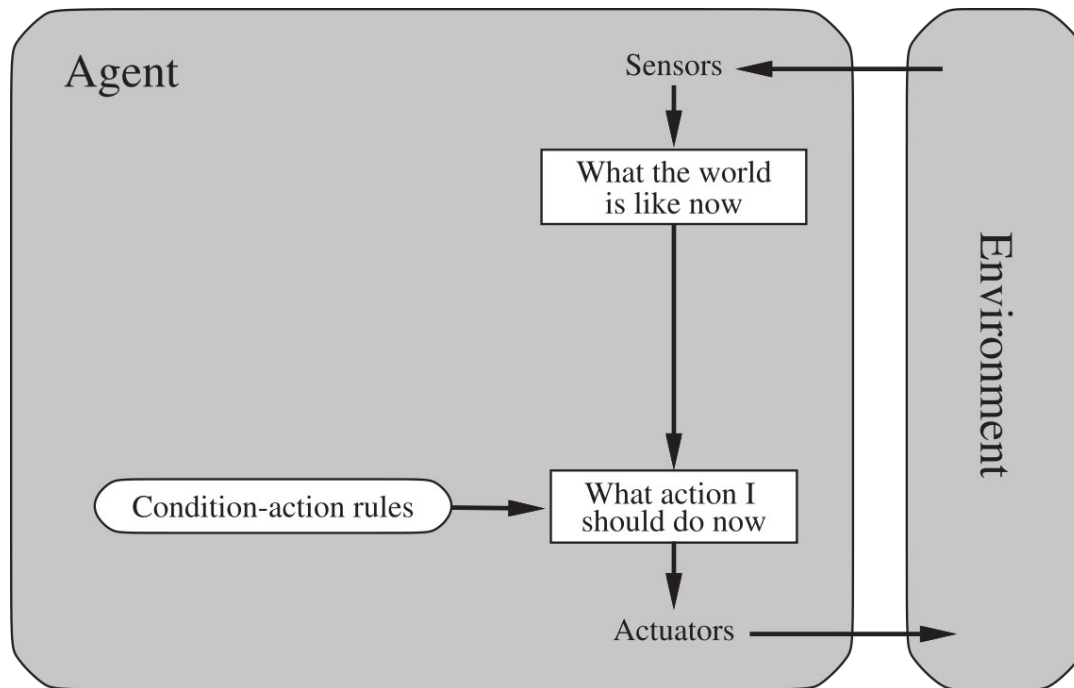
Agente

- **Agente = architettura + programma**
 - **Architettura** = specifica degli elementi strutturali e funzionali
 - **Programma** = funzione che mette in relazione percezioni e azioni
- Distinguiamo fra funzione agente e programma agente:
 - **Funzione agente**: ha come input la sequenza percettiva (la storia delle percezioni)
 - **Programma agente**: ha come input la percezione corrente

Tipologie di agente

- Agenti reattivi (semplici)
- Agenti reattivi basati su modello
- Agenti guidati dagli obiettivi (goal-driven)
- Agenti guidati dall'utilità (utility-driven)
- Agenti capaci di apprendere

Agenti reattivi (semplici)



Basano la scelta dell'azione sulla sola percezione corrente

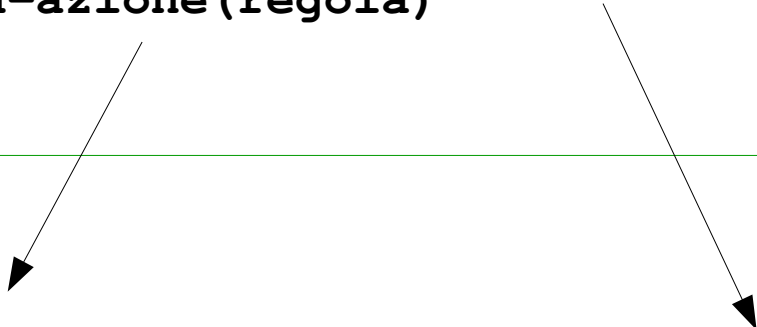
Il programma codifica delle **regole condizione-azione**
Agente specifico, scritto ad hoc per un certo problema

ESEMPIO:

```
Aspirapolvere-reattivo (posizione, stato) return azione
{
  if (stato = sporco) then return aspira
  else if (posizione = A) then return destra
  else if (posizione = B) then return sinistra
}
```

Agente reattivo semplice e generale

```
Agente-reattivo-semplice (percezione) return azione
{
  stato ← interpreta(percezione)
  regola ← individua-regola(stato, regole)
  azione ← regola-azione(regola)
  return azione
}
```



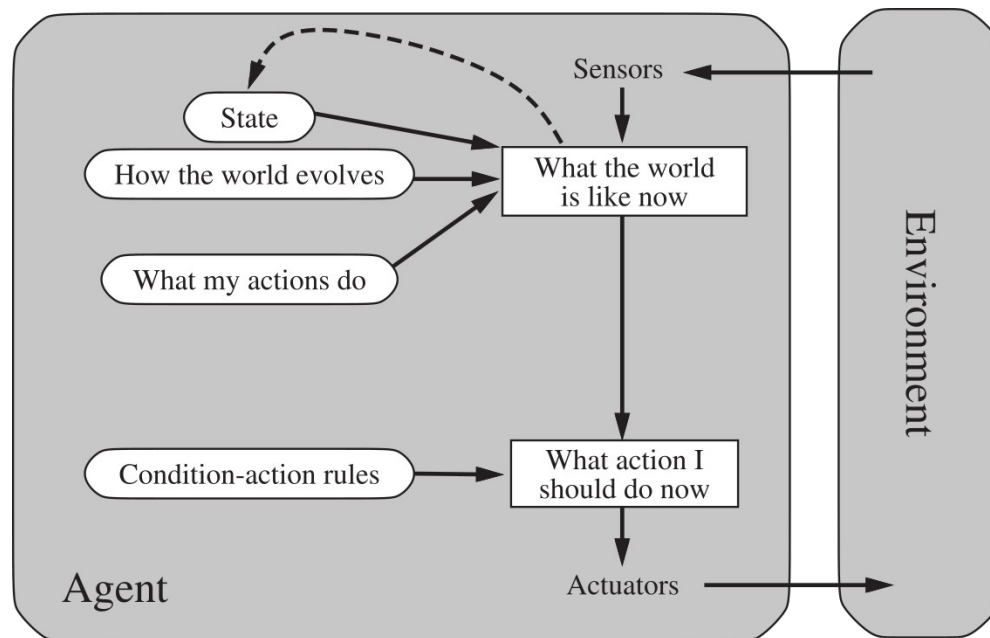
La percezione è usata per identificare lo stato; lo stato identifica la regola da applicare; la regola dice quale azione scegliere per l'esecuzione

l'insieme delle regole determina il comportamento dell'agente

Agenti reattivi semplici: limiti

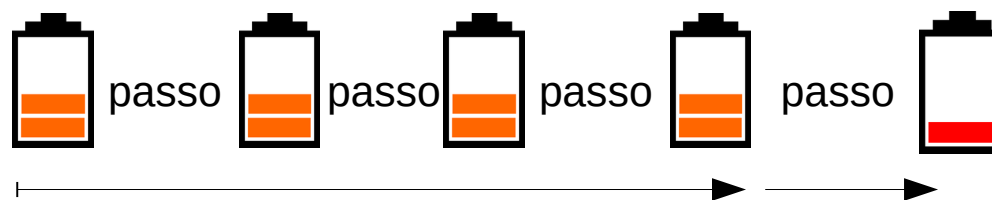
- Funzionano in ambienti **completamente osservabili**
- Supponiamo che l'aspiratore **manchi del sensore di posizione**:
 - se la stanza in cui si trova è sporca aspira,
 - se è pulita cosa deve fare? Per sapere dove andare deve sapere dove si trova
 - Si possono creare **loop infiniti** ([pulito] destra[pulito] destra [pulito] ...)
 - Per evitare i loop infiniti occorre introdurre **comportamenti random** (casuali)
- l'azione deve essere determinabile dalla **sola percezione corrente**
 - Se l'aspirapolvere potesse attivare un solo sensore per volta, di nuovo non saprebbe sempre come comportarsi (immaginate: sensore sporco/pulito dà pulito ma dove andare? Sensore posizione dice dove andare ma la stanza in cui è il robot è da aspirare?)

Agenti reattivi basati su modello



In caso di osservabilità parziale si possono usare agenti che hanno un modello del mondo: conoscenza su come il mondo evolve e sugli effetti delle azioni

**Modello +
Sequenza percettiva +
Storia delle azioni**



Esempio:

So che quando la carica della batteria diventa media, rimane in questo stato per tre passi poi diventa rossa. Conoscenza e memoria delle percezioni eliminano il non-determinismo

Agente reattivo basato su modello

```
Agente-reattivo-con-modello (percezione) return azione
{
  stato ← aggiorna-stato(stato, azione, percezione)
  regola ← individua-regola(stato, regole)
  azione ← regola-azione(regola)
  return azione
}
```

Aggiorna-stato: determina lo stato corrente sulla base (1) della percezione corrente, (2) dell'azione più recente, (3) dello stato precedente e (4) di un modello del mondo che cattura come il mondo evolve.

Avere un modello permette di **prevedere gli effetti delle azioni** e di **scegliere** l'azione da eseguire anche sulla base di questa informazione

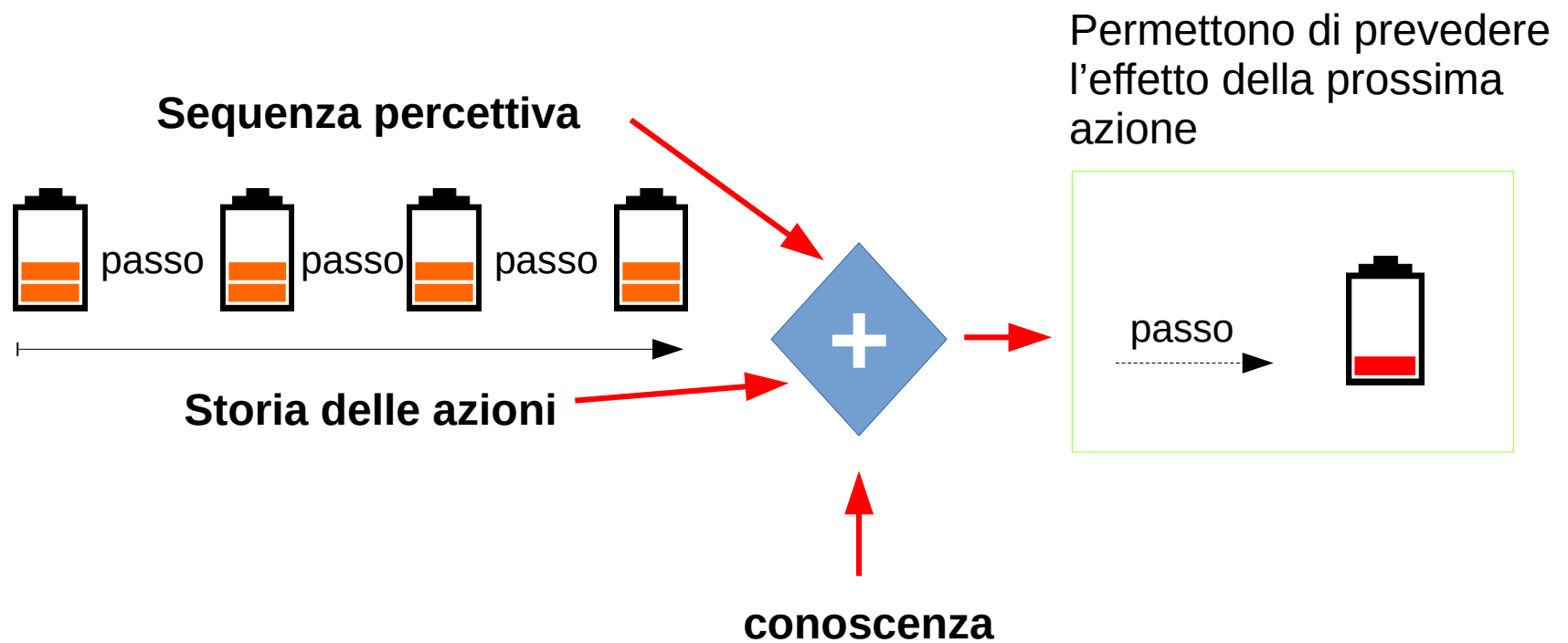
Su cosa basare le scelte?

Agenti reattivi basati su modello

Modello = tre passi consumano una tacca di batteria

Sequenza percettiva = 2 tacche, 2 tacche, 2 tacche

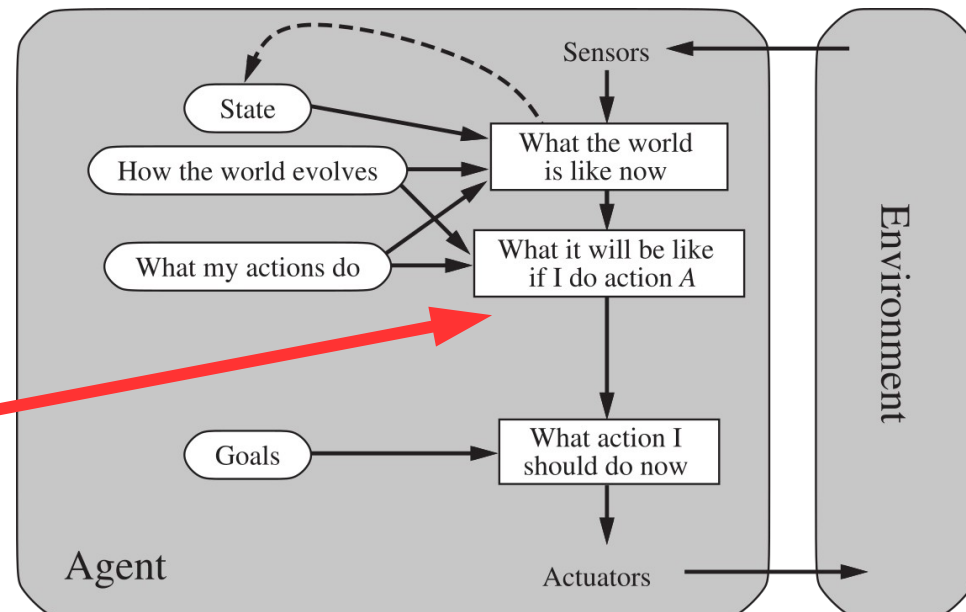
Storia delle azioni = passo, passo, passo



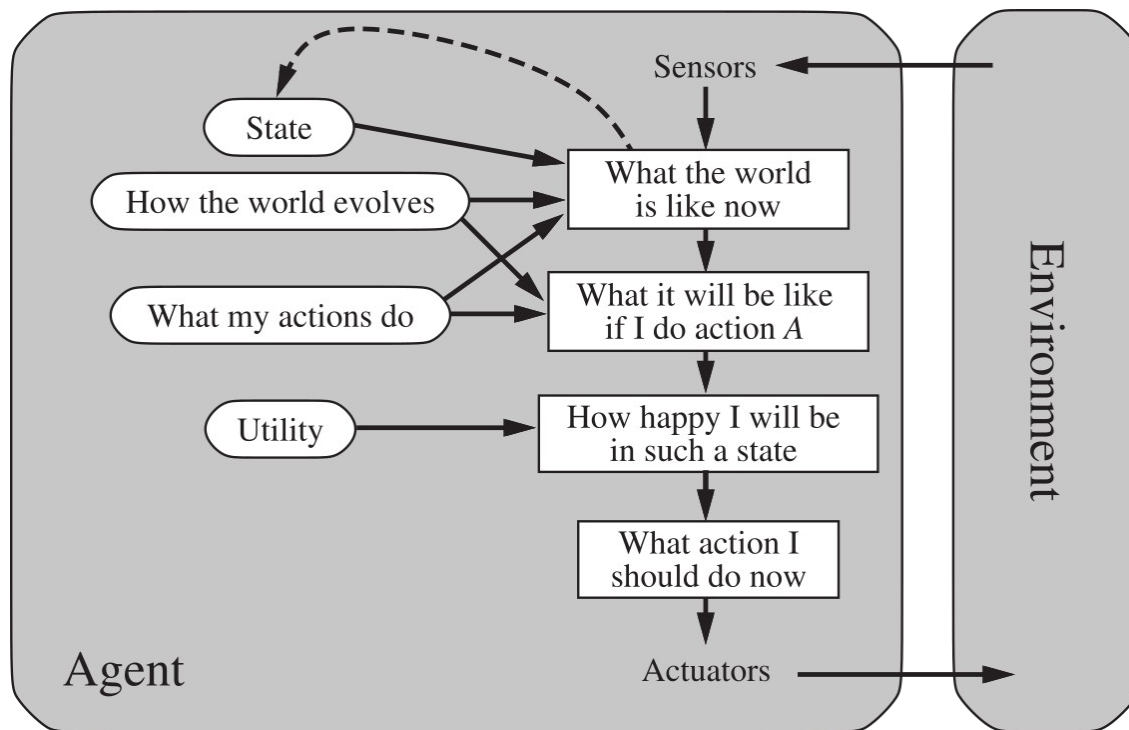
Agenti basati su obiettivi

- L'agente sceglie l'azione da eseguire sulla base dei propri **obiettivi**, cioè l'azione deve avvicinarlo ai suoi obiettivi o farglieli raggiungere
- La decisione può coinvolgere il solo passo successivo (passo singolo) o guardare in avanti per più passi (piano)
- **VANTAGGIO**: cambiando gli obiettivi dell'agente posso fargli realizzare diversi comportamenti

Ragionamento ipotetico
L'agente "simula" nella propria "mente" l'effetto delle azioni per capire se e quanto l'avvicinano all'obiettivo



Agenti basati sull'utilità

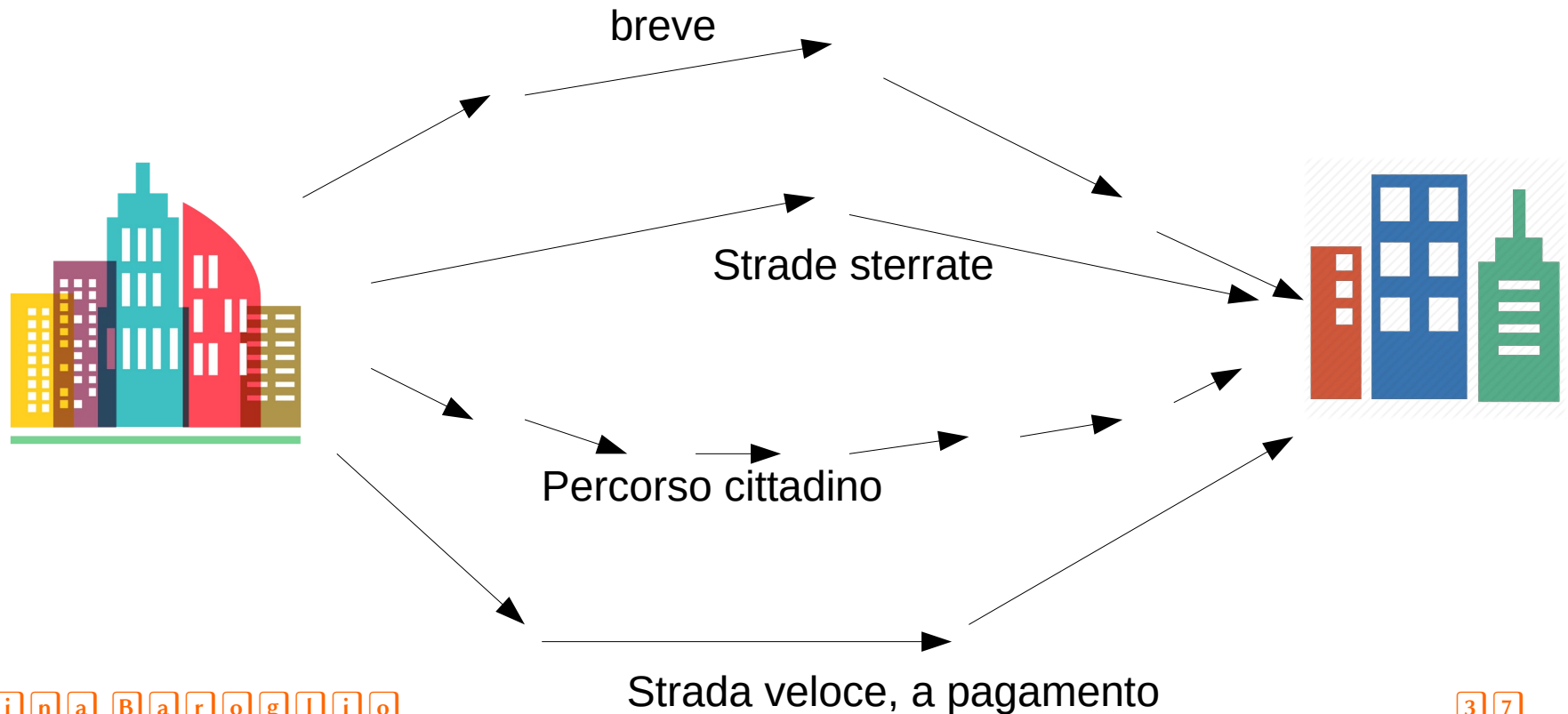


Alla nozione di obiettivo si aggiunge una misura della prestazione, cioè della bontà delle possibili scelte calcolata da una funzione metrica

Funzioni di utilità, tengono conto di fattori quali ad esempio: *costo, velocità, difficoltà attuativa, tempo, risorse utilizzate*

Utilità: esempio

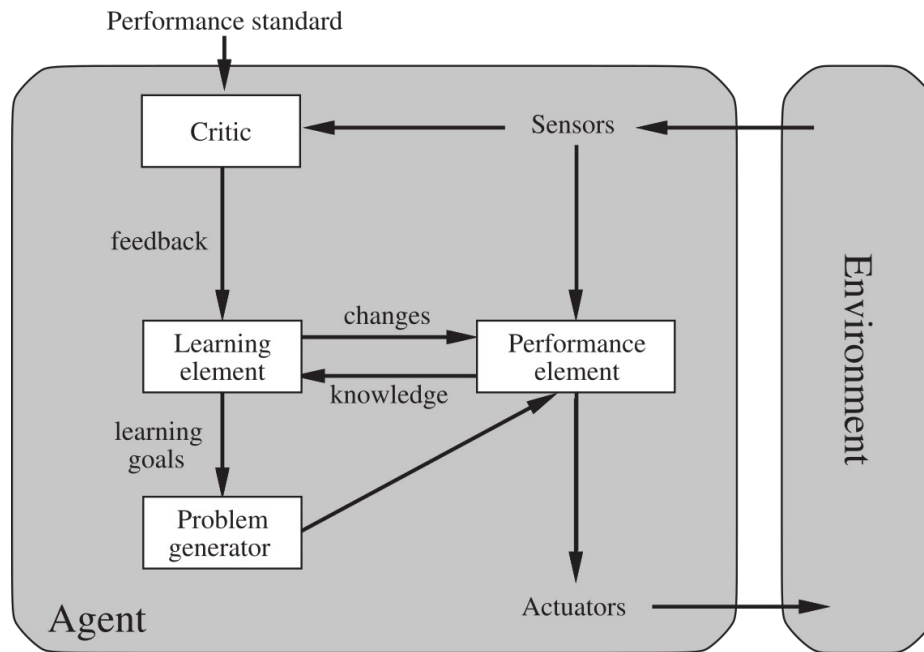
- Per raggiungere una località sono disponibili diverse soluzioni (diversi percorsi)
- Il problema non è solo determinare una soluzione ma determinarne una che ci “**rende felici**”



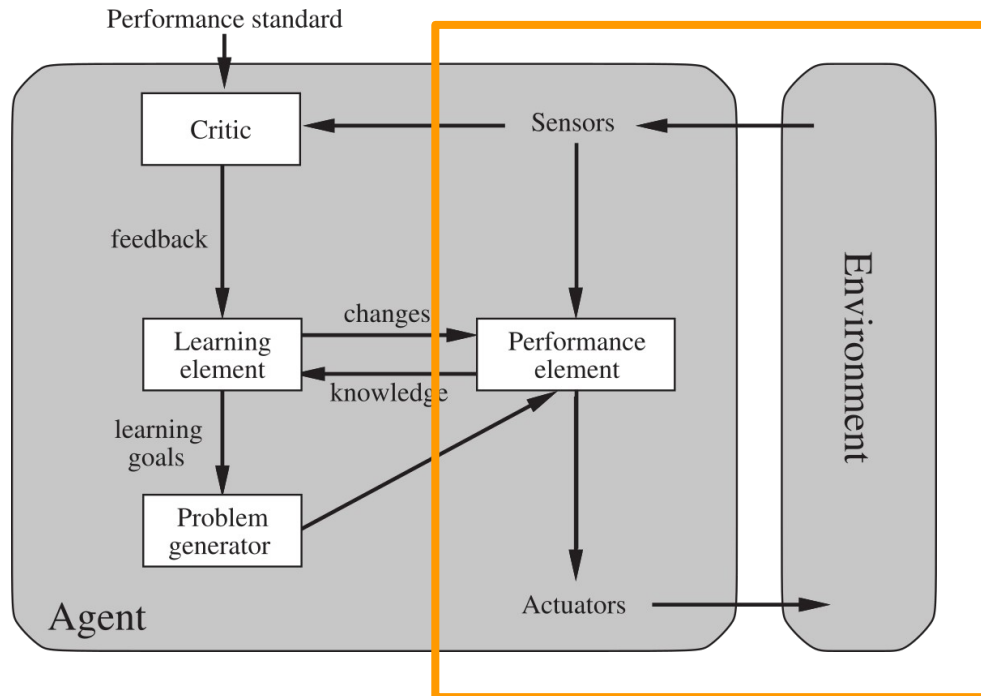
Goal e utilità

- **Obiettivo**: è un particolare stato che si vuole raggiungere, normalmente descritto in termini di proprietà
- **Utilità**: è una funzione che, dato uno stato o una sequenza di stati, restituisce una misura
- **L'agente ha un duplice problema**:
(1) raggiungere l'obiettivo (2) massimizzando l'utilità
- **Si distingue fra soluzioni e buone soluzioni**: utilità usata come guida per determinare una buona soluzione

Agenti che apprendono

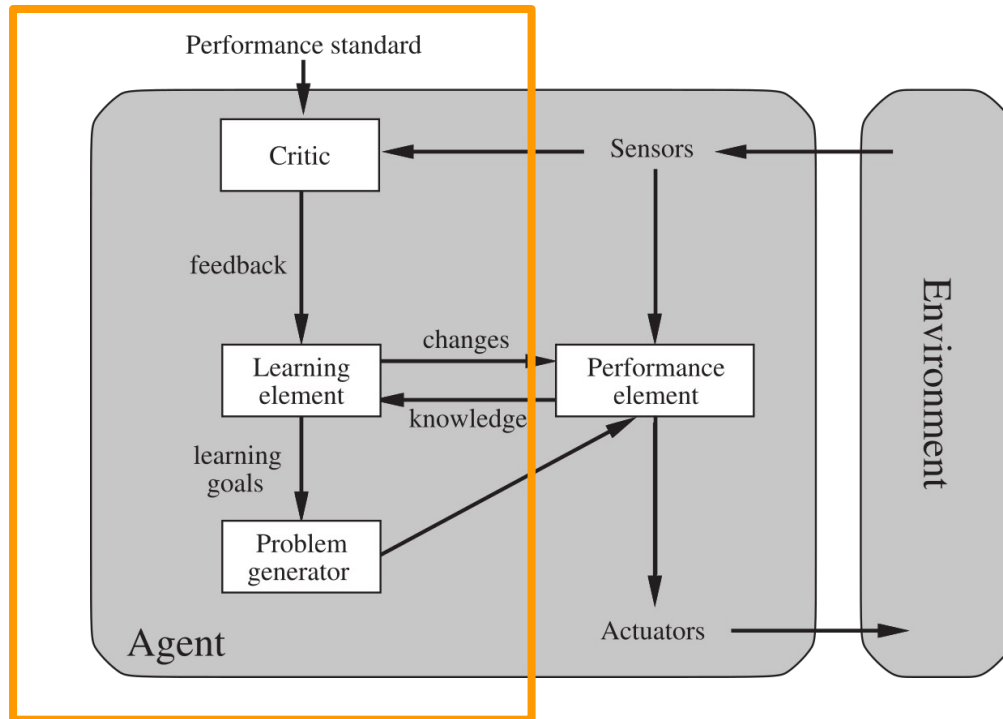


Agenti che apprendono



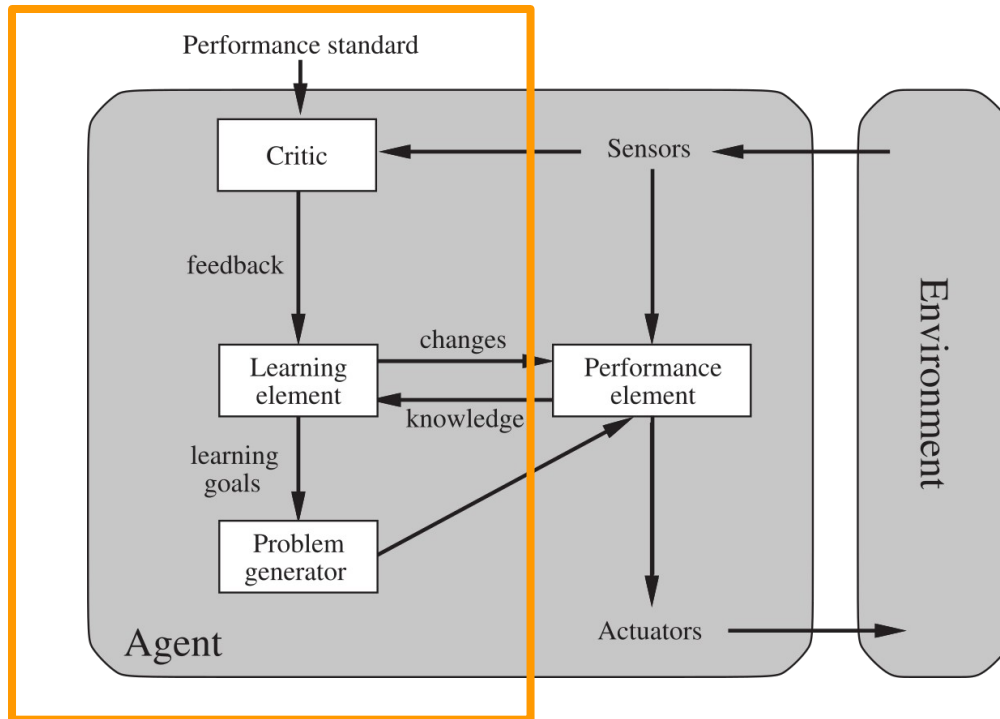
Questa parte rappresenta l'agente come visto finora: percepisce, decide quale azione eseguire, la attua

Agenti che apprendono



- Questa parte è invece deputata dell'apprendimento ed è costituita da 3 elementi:
- 1) Un **critico**, che valuta il livello di prestazione dell'agente decidendo se è il caso di attivare l'apprendimento;
 - 2) Un **modulo di apprendimento** che modifica la conoscenza dell'agente
 - 3) Un **generatore di problemi** che causa l'esecuzione di azioni esplorative, il cui fine è esporre l'agente a nuove esperienze

Agenti che apprendono



Non facciamo assunzioni su quali segnali e quali tecniche l'agente usi per apprendere e quindi per modificare il proprio comportamento migliorandone le prestazioni

Questa parte è invece deputata dell'apprendimento ed è costituita da 3 elementi:

- 1) Un critico, che valuta il livello di prestazione dell'agente decidendo se è il caso di attivare l'apprendimento;
- 2) Un modulo di apprendimento che modifica la conoscenza dell'agente
- 3) Un generatore di problemi che causa l'esecuzione di azioni esplorative, il cui fine è esporre l'agente a nuove esperienze

Multiagent System (MAS)

FACOLTATIVO

- Un sistema multiagente è costituito da un insieme di agenti che operano in uno stesso ambiente
- Gli agenti possono competere o collaborare nell'uso delle risorse fornite dall'ambiente e nel perseguimento dei propri obiettivi
- **Come permettere l'interazione?**

Protocolli di interazione

FACOLTATIVO

- Un **protocollo di interazione** è uno schema che permette di coordinare due o più agenti, tipicamente sfruttando uno **scambio di messaggi** come strumento di coordinazione
- Gli agenti possono essere implementati **indipendentemente** ed essere **eterogenei** ma possono interagire con successo se:
 - 1) Condividono **un'ontologia** del dominio del discorso
 - 2) Condividono un **linguaggio** di comunicazione
 - 3) Condividono un **protocollo** di interazione

FIPA interaction protocols

FACOLTATIVO

- **FIPA: Foundation for Intelligent Physical Agents**
- A inizio anni 2000 ha definito una semantica formale per gli **atti comunicativi** e ha standardizzato un insieme di **protocolli di interazione** fra i quali:
 - Contract Net Interaction Protocol
 - Brokering Interaction Protocol
 - Query Interaction Protocol
 - ...

FIPA Speech Acts

FACOLTATIVO

- FIPA ha standardizzato la semantica di molti atti comunicativi, fra i quali AcceptProposal, Agree, Cancel, Call for proposal, Confirm, Disconfirm, Failure, Inform, ...
- Esempio di atto comunicativo di tipo Request**
Agent i requests agent j to deliver a box to a certain location

- (request**

- :sender (agent-identifier :name i)
 - :receiver (set (agent-identifier :name j))
 - :content

- "((action (agent-identifier :name j)
 (deliver box017 (loc 12 19))))"

→ Ontologia del dominio

- :protocol **fipa-request**
 - :language **fipa-sl**
 - :reply-with order567)

FIPA Speech Acts, continua

FACOLTATIVO

- **j answers that it agrees to the request but it has low priority**

- **(agree**
:sender (agent-identifier :name j)
:receiver (set (agent-identifier :name i))
:content
"((action (agent-identifier :name j)
 (deliver box017 (loc 12 19)))
 (priority order567 low))"
:in-reply-to **order567**
:protocol **fipa-request**
:language **fipa-sl**)

Esempio di protocollo

FACOLTATIVO

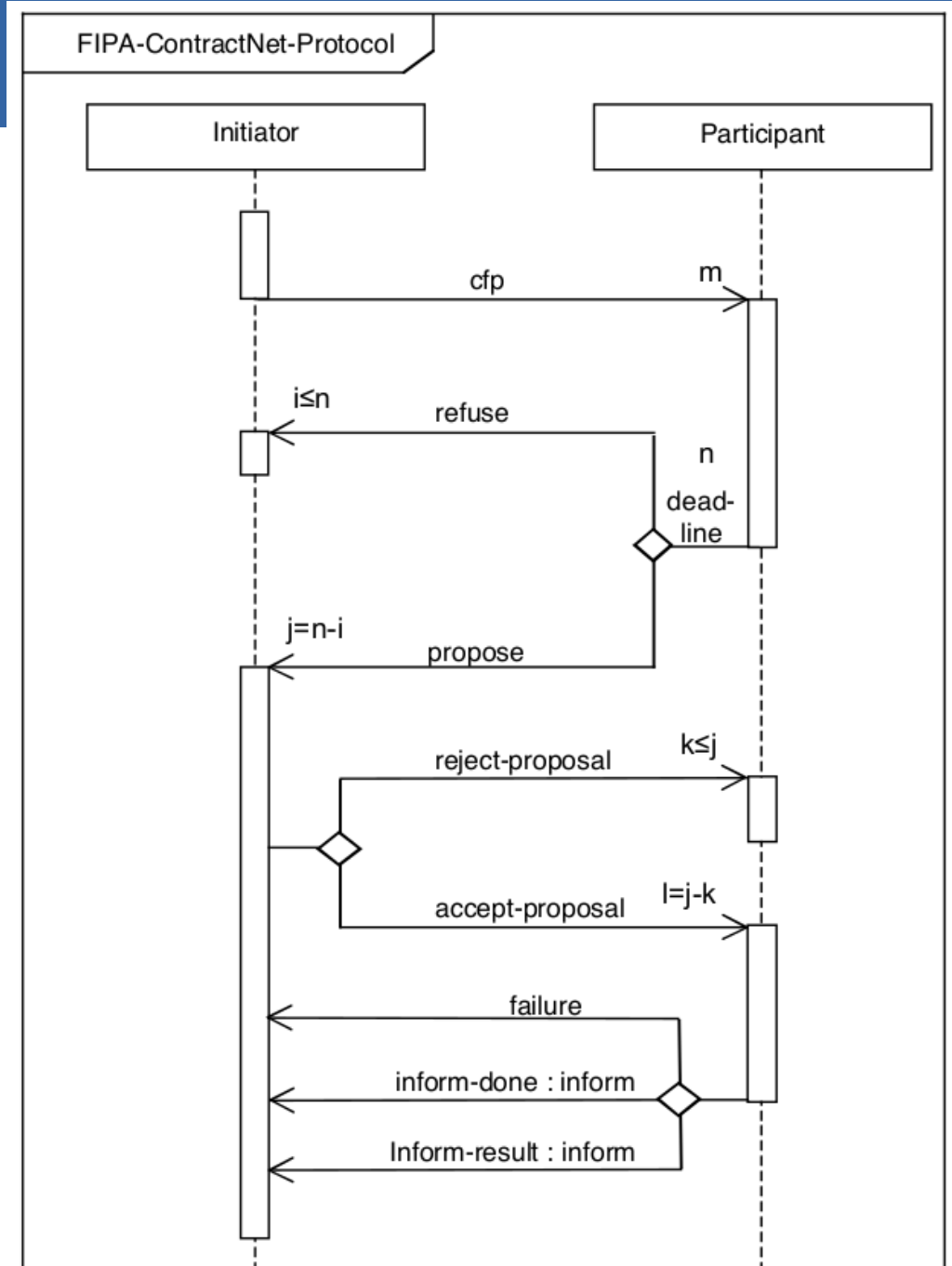


Figure 1: FIPA Contract Net Interaction Protocol

Esempio di protocollo

FACOLTATIVO

Ruoli

Timeline

Comunicazioni

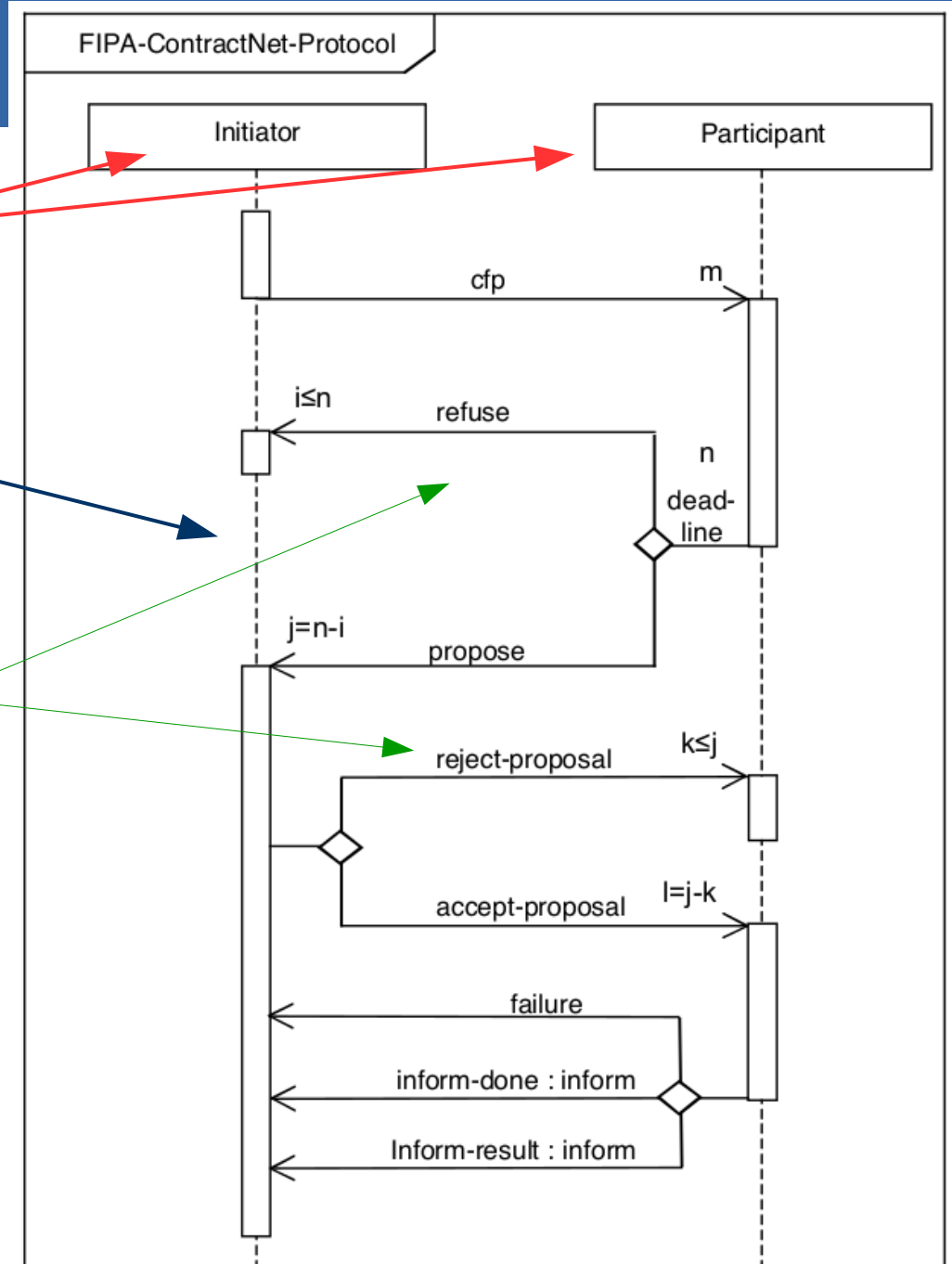


Figure 1: FIPA Contract Net Interaction Protocol

Multiagent Systems: Software

FACOLTATIVO

- JADE (<http://jade.tilab.com>)
- Jason (<http://jason.sourceforge.net/wp/>)
- CartAgO (<http://cartago.sourceforge.net/>)
- JaCaMo (<http://jacamo.sourceforge.net/>)