

Agenti intelligenti

Alessio Micheli

a.a. 2022/2023

Credits: Maria Simi

Russell-Norvig

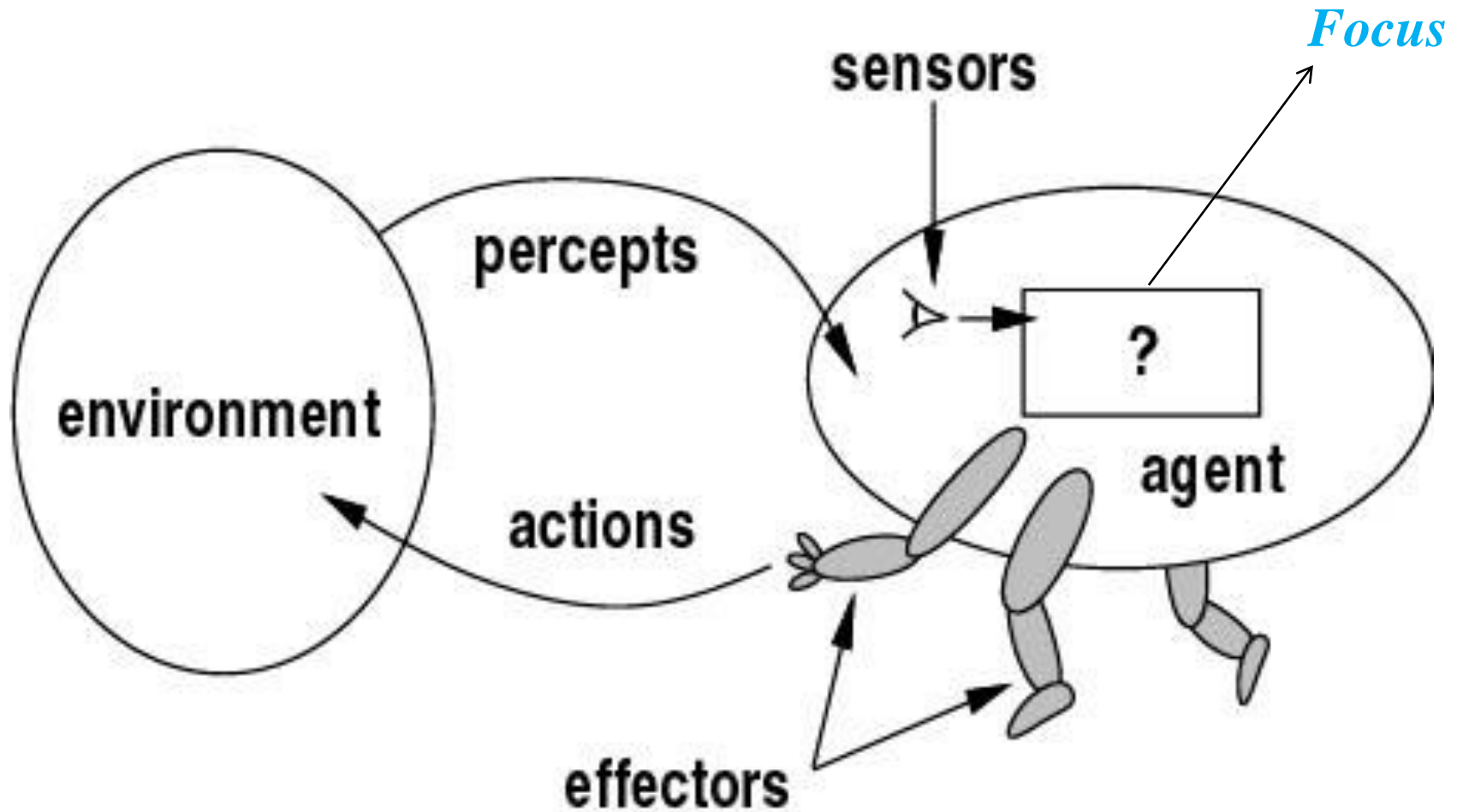
IA — “ri”considerazioni

- Non collezione di tecniche per risolvere problemi specifici
- Ma vertice e fronte del progresso dei metodi/sistemi informatici (e.g. algoritmica, logica, ottimizzazione) per fornire metodologie sistematiche per dotare le macchine di comportamenti intelligenti/ razionali (cfr. *ALMA*) in problemi generali *difficili*
- Iniziamo con l'inquadramento degli **agenti**

Agenti intelligenti — visione nel corso

- L'approccio “moderno” all'IA: costruzione di *agenti intelligenti*
- La visione ad agenti ci offre un quadro di riferimento e una prospettiva diversa all'analisi dei sistemi software
 - E “comoda” per trattare sistemi razionali (uniformità)
 - Vedremo schemi di agenti contenitori di funzionalità studiate nel resto del corso
- Primo obiettivo: agenti per risoluzione di problemi vista come **ricerca** in uno **spazio di stati** (*problem solving*).

Agenti secondo ALMA



Ciclo *percezione- azione*



Caratteristiche degli agenti

Rispetto ad un «modulo sw»

- Gli agenti sono **situati**
 - ricevono *percezioni* da un ambiente
 - agiscono sull'ambiente mediante *azioni* (attuatori)
- Gli agenti hanno **abilità sociale**
 - sono capaci di comunicare, collaborare, difendersi da altri agenti
- Gli agenti hanno **credenze, obiettivi, intenzioni ...**
- Gli agenti sono **embodied**: hanno un **corpo** , fino a considerare i meccanismi delle **emozioni**

Percezioni e azioni

- Percezione: input da sensori
- *Sequenza percettiva*: storia completa delle percezioni
- La scelta dell'azione è *funzione unicamente* della sequenza percettiva (ma non da qualcosa che non abbia percepito)
- *Funzione agente*: definisce l'azione da compiere per ogni sequenza percettiva (descrive completamente l'agente).

$$\text{Sequenza percettiva} \xrightarrow{f} \text{Azione}$$

- Implementata da un *programma agente*
- Compito (IA): progettare il programma agente

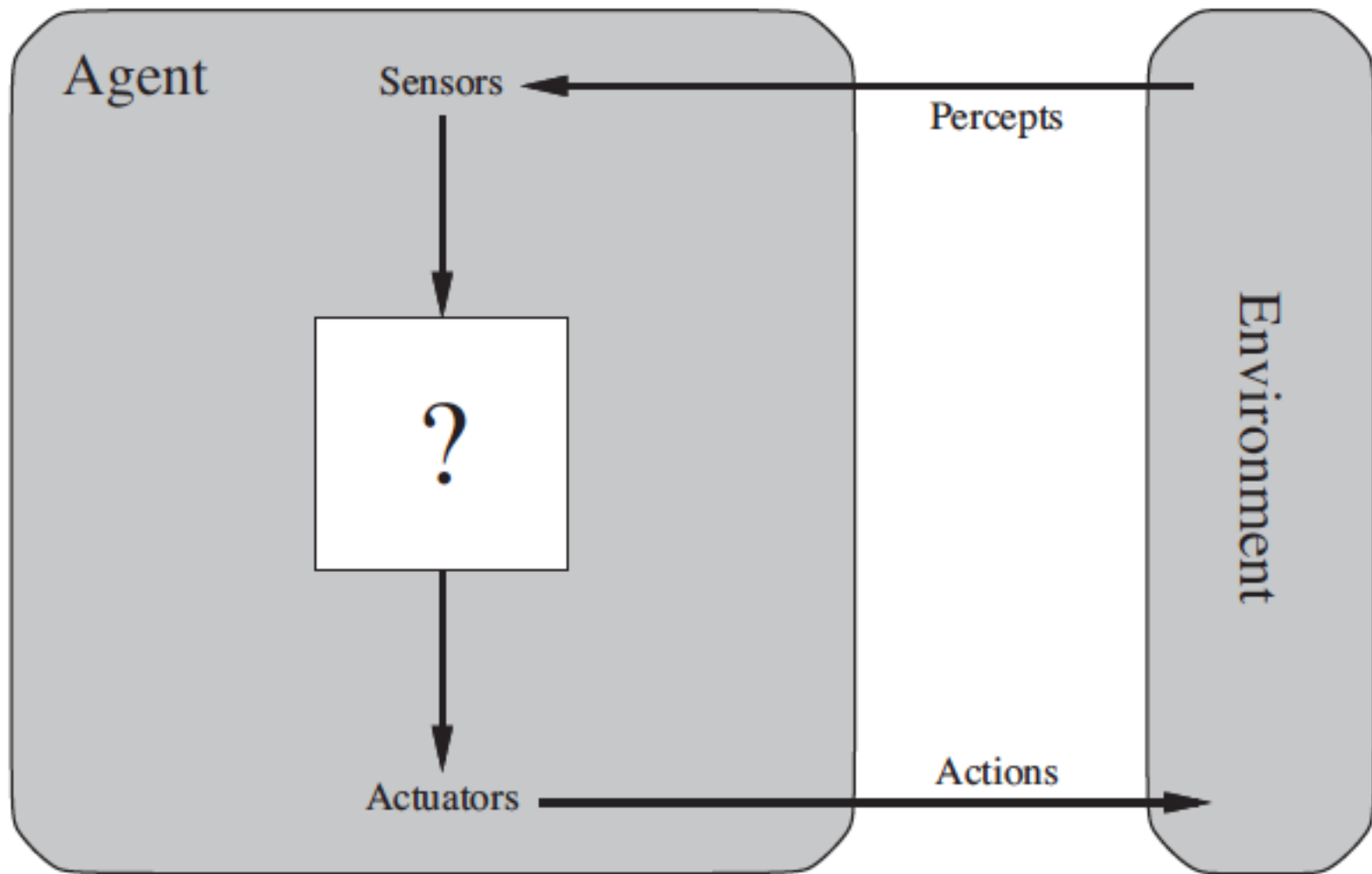
Sommario (cap. 2, ALMA)

- **Agenti razionali**
- Proprietà degli ambienti
- La struttura degli agenti
 - Agenti reattivi semplici
 - Agenti basati su modello
 - Agenti con obiettivo
 - Agenti con funzione di utilità
 - Agenti che apprendono

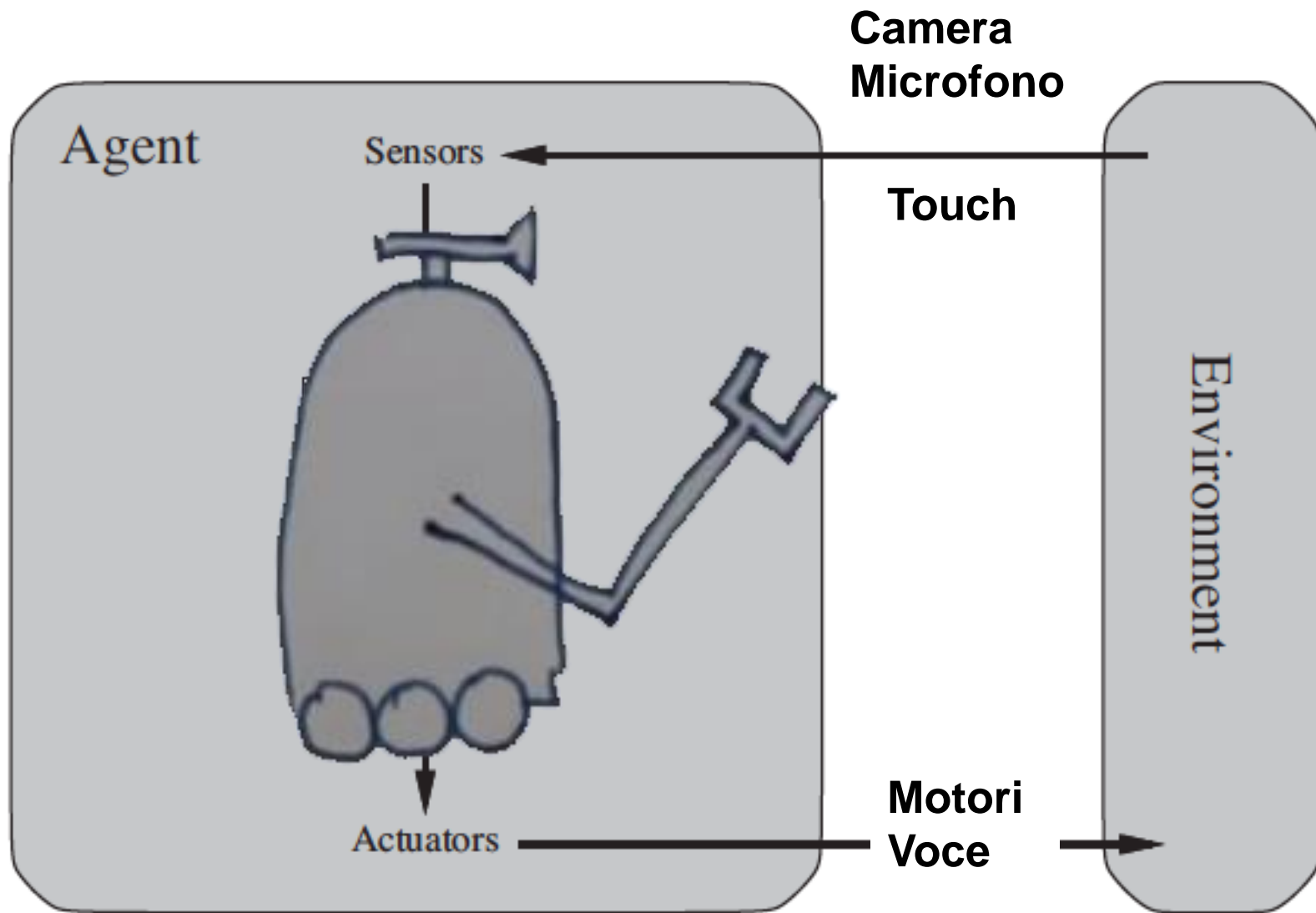
Rapida overview



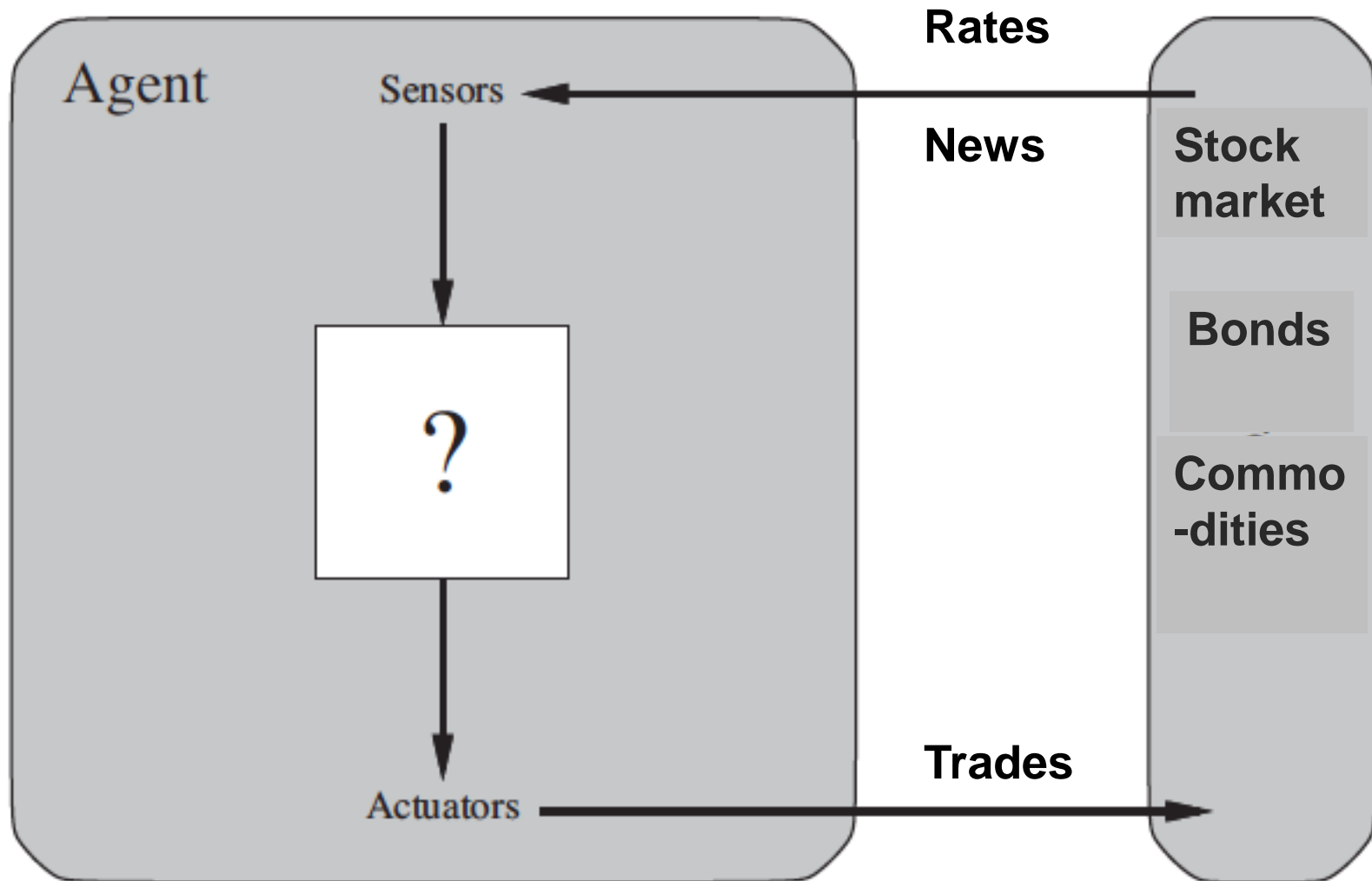
Agente e ambiente (architettura astratta)



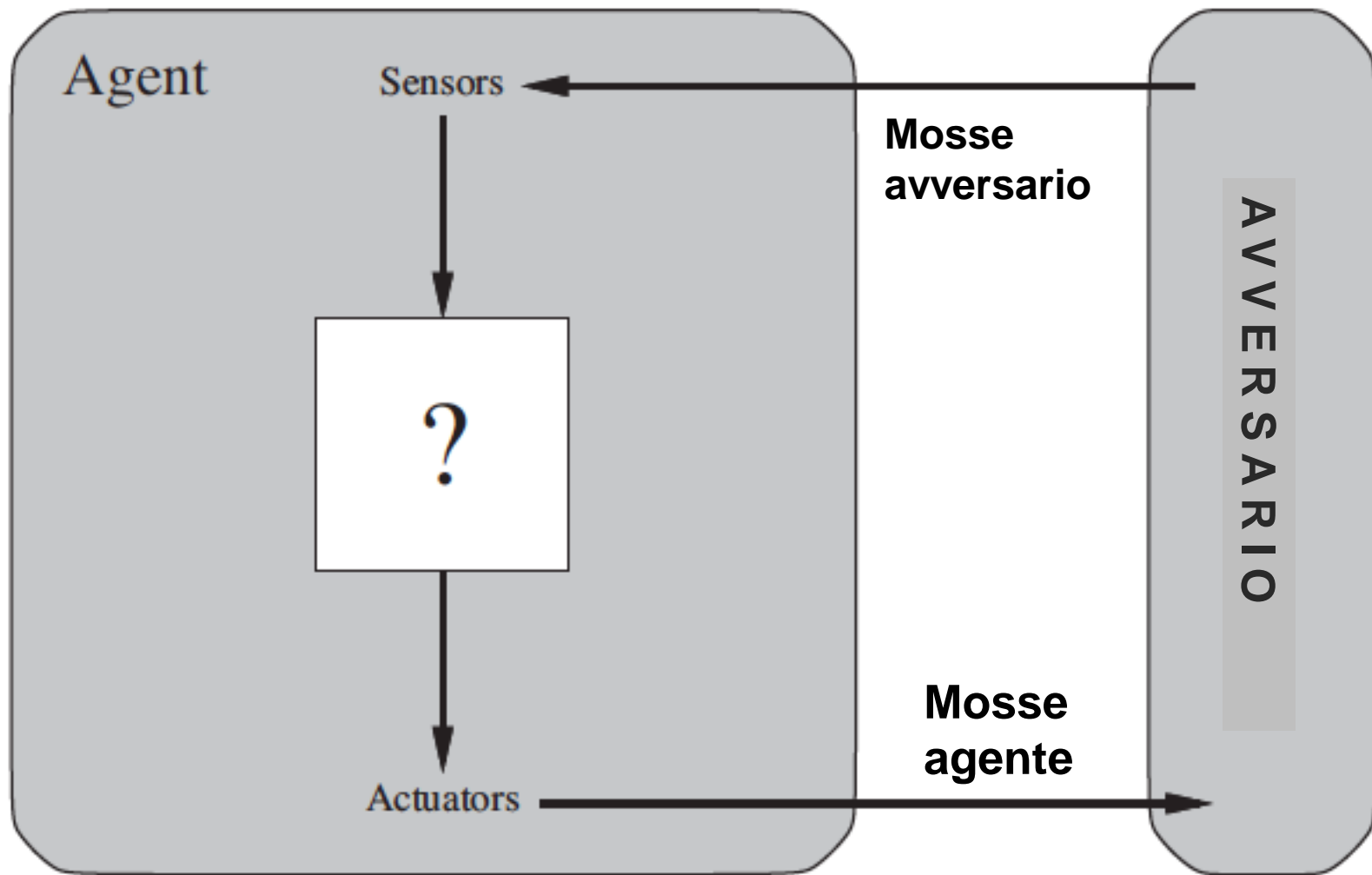
Agente robotico



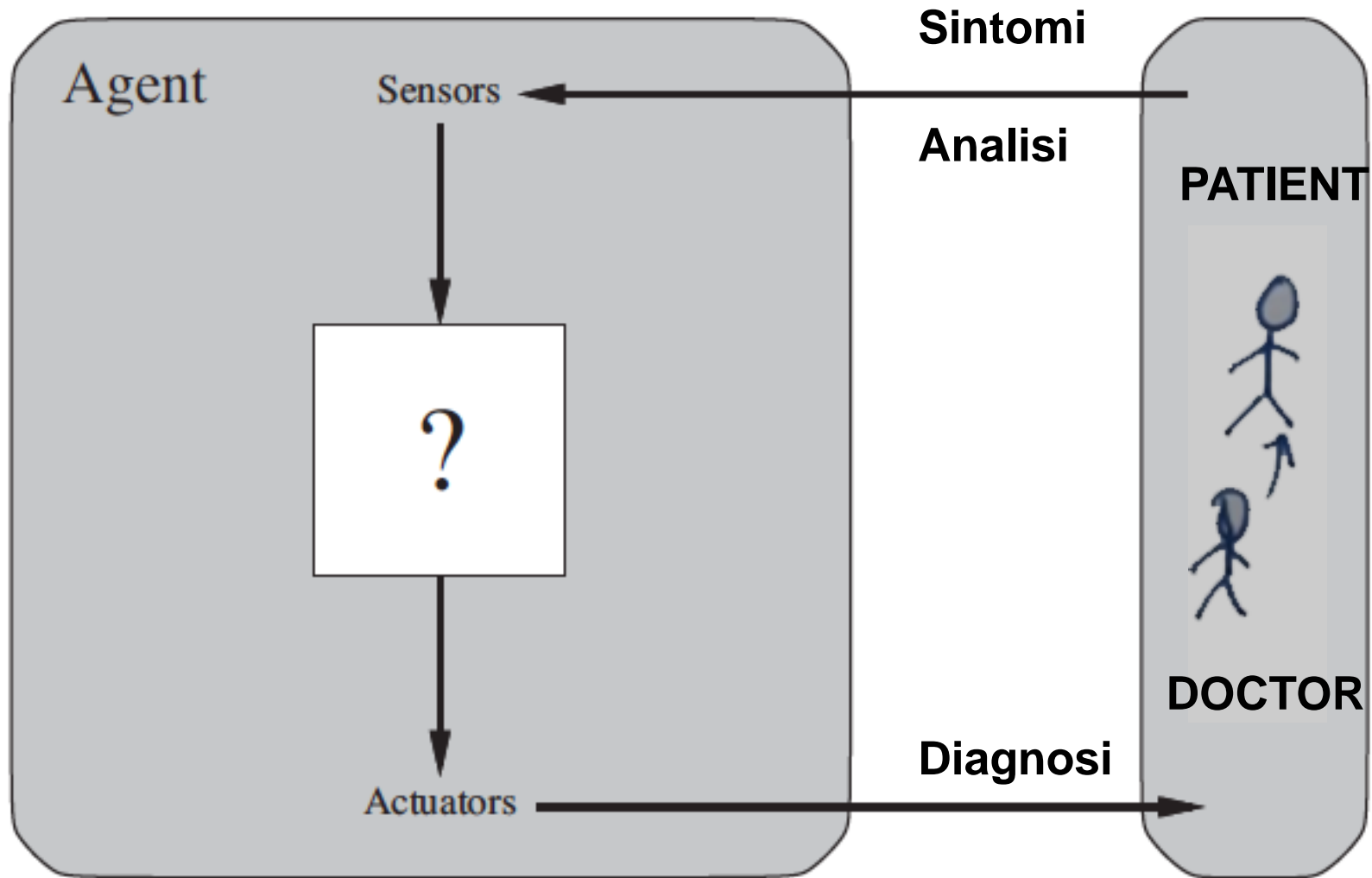
Agente finanziario (trading agent)



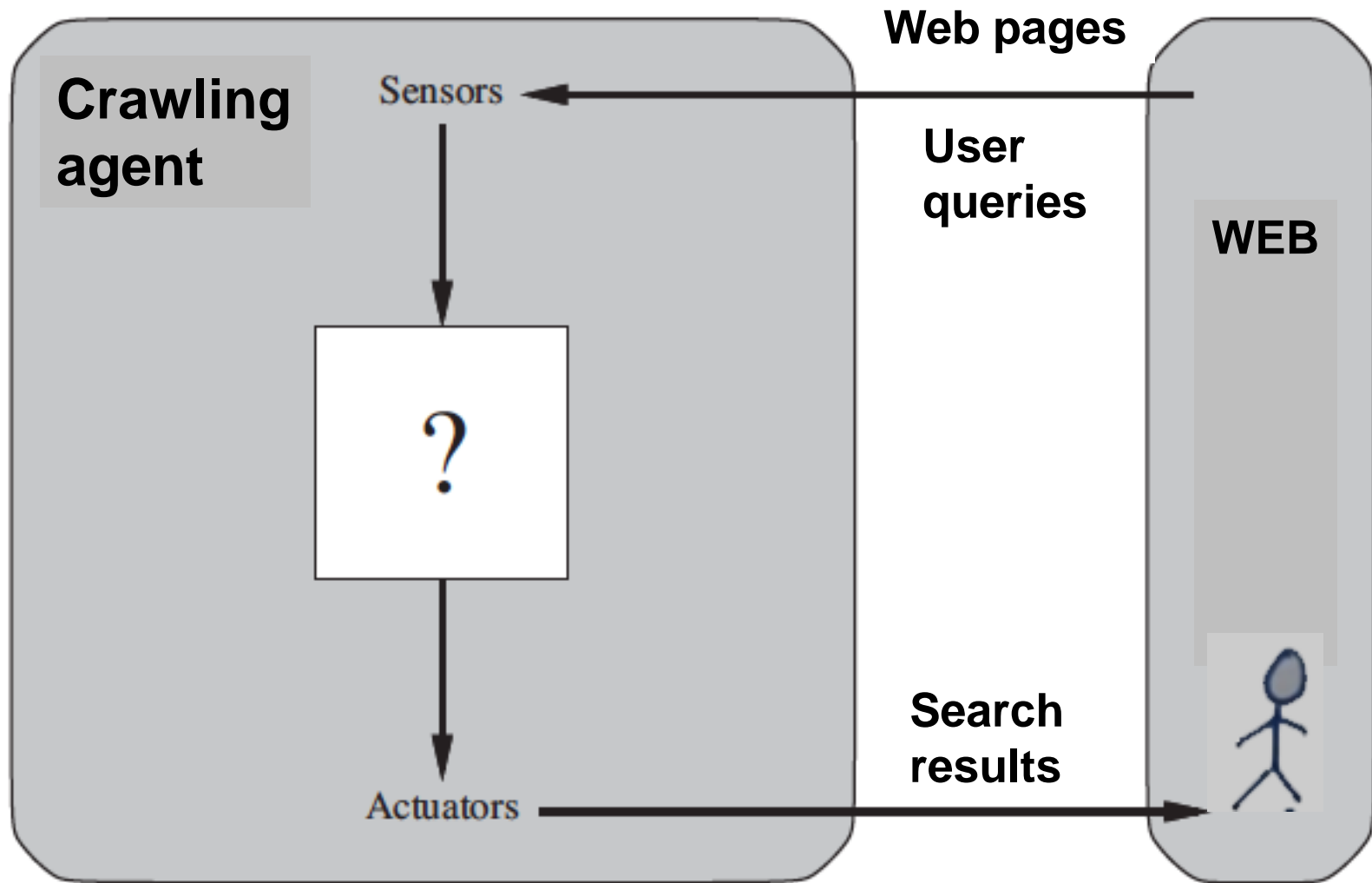
Game agent



Agente diagnostico in medicina



IA e web: motore di ricerca



Agenti razionali

- Un *agente razionale* interagisce con il suo ambiente in maniera “efficace” (fa la cosa “giusta”).
- Serve un criterio di *valutazione* oggettivo dell'effetto delle azioni dell'agente (della sequenza di stati dell'ambiente)

e.g. costo minimo di un cammino alla soluzione

Valutazione della prestazione



- **Misura di prestazione**

- Esterna (come vogliamo che il mondo evolva?)
- Scelta dal progettista a seconda del problema considerando l'effetto desiderato sull'ambiente
- (possibile) Valutazione su ambienti diversi

Agente razionale: definizione

- La razionalità è relativa a (dipende da):
 - la misura di prestazioni
 - le conoscenze pregressa dell'ambiente
 - le percezioni presenti e passate (seq. percettiva)
 - le capacità dell'agente (azioni possibili)
- *Agente razionale*: per ogni sequenza di percezioni compie l'azione che massimizza il valore atteso della misura delle prestazioni, considerando le sue percezioni passate e la sua conoscenza pregressa.

Razionalità non onniscienza

e.g. Memento: La scelta dell'azione è *funzione unicamente* della sequenza percettiva



- Non si pretendono perfezione e conoscenza del “futuro”, ma massimizzare il risultato *atteso*
- Ma potrebbe essere necessarie azioni di acquisizione di informazioni o esplorative

Razionalità non onnipotenza

- Le capacità dell'agente possono essere limitate



Razionalità e apprendimento

- Raramente tutta la conoscenza sull'ambiente può essere fornita “a priori” (dal programmatore).
- L'agente razionale deve essere in grado di modificare il proprio comportamento con l'esperienza (le percezioni passate).
- Può migliorare esplorando, apprendendo, aumentando autonomia per operare in ambienti differenti o mutevoli (da vedere più avanti nel corso...)

Agenti autonomi



- *Agente autonomo*: un agente è autonomo nella misura in cui il suo comportamento dipende dalla sua capacità di ottenere esperienza (e non dall'aiuto del progettista)
- Un agente il cui comportamento fosse determinato solo dalla sua conoscenza *built-in* (pregressa), sarebbe non autonomo e poco flessibile



Sommario (cap. 2, ALMA)

- Agenti razionali
- **Proprietà degli ambienti**
- La struttura degli agenti
 - Agenti reattivi semplici
 - Agenti basati su modello
 - Agenti con obiettivo
 - Agenti con funzione di utilità
 - Agenti che apprendono

Rapida overview



Ambienti

- Definire un problema per un agente significa caratterizzare l'ambiente in cui l'agente opera (ambiente operativo).
- Descrizione PEAS dei problemi
 - **P** erformance | prestazione
 - **E** nvironment | ambiente
 - **A** ctuators | attuatori
 - **S** ensors | sensori

Agente guidatore di taxi

Prestazione	Ambiente	Attuatori	Sensori
Arrivare alla destinazione, sicuro, veloce, ligio alla legge, viaggio confortevole, minimo consumo di benzina, profitti massimi	Strada, altri veicoli, pedoni, clienti	Sterzo, acceleratore, freni, frecce, clacson, schermo di interfaccia o sintesi vocale	Telecamere, sensori a infrarossi e sonar, tachimetro, GPS, contachilometri, accelerometro, sensori sullo stato del motore, tastiera o microfono

Formulazione PEAS dei problemi

Esercizio



Problema	P	E	A	S
Diagnosi medica	Diagnosi corretta	Pazienti, ospedale	Domande, suggerimenti test, diagnosi	Sintomi, Test clinici, risposte paziente
Analisi immagini	% img/zone correttamente classificate	Collezione di fotografie	Etichettatore di zone nell'immagine	Array di pixel
Robot “selezionatore”	% delle parti correttamente classificate	Nastro trasportatore	Raccogliere le parti e metterle nei cestini	Telecamera (pixel di varia intensità)
Giocatore di calcio	Fare più goal dell'avversario	Altri giocatori, campo di calcio, porte	Dare calci al pallone, correre	Locazione pallone altri giocatori, porte
Broker finanziario				

Proprietà dell'ambiente-problema

- Completamente/parzialmente osservabile
- Agente singolo/multi-agente
- Deterministico/stocastico/non deterministico
- Episodico/sequenziale (esiste storia?)
- Statico/dinamico
- Discreto/continuo



Dubbi?: slide successive...



Osservabilità

- Ambiente **completamente osservabile**
 - L'apparato percettivo è in grado di dare una conoscenza completa dell'ambiente o almeno tutto quello che serve a decidere l'azione
 - Non c'è bisogno di mantenere uno stato del mondo (esterno)
- Ambiente **parzialmente osservabile**
 - Sono presenti limiti o inaccuratezze dell'apparato sensoriale.



Ambiente singolo/multiagente

- Distinzione agente/non agente
 - Il mondo può anche cambiare per *eventi*, non necessariamente per azioni di agenti.
- Ambiente multi-agente competitivo (schacchi)
 - Comportamento randomizzato (è razionale)
- Ambiente multi-agente cooperativo (o benigno)
 - Stesso obiettivo
 - Comunicazione



Predicibilità

- **Deterministico**

- Se lo stato successivo è completamente determinato dallo stato corrente e dall'azione. Esempio: scacchi

- **Stocastico**

- Esistono elementi di incertezza con associata probabilità. Esempi: guida, tiro in porta

- **Non deterministico**

- Si tiene traccia di più stati possibili risultato dell'azione (ma non in base ad una probabilità)



Episodico/sequenziale

- Episodico

- L'esperienza dell'agente è divisa in episodi atomici indipendenti (es. partite diverse)
- In ambienti episodici non c'è bisogno di pianificare

- Sequenziale

- Ogni decisione influenza le successive (es. scacchi in una partita)



Statico/dinamico

■ Statico

- Il mondo non cambia mentre l'agente decide l'azione (es. Cruciverba)

■ Dinamico

- Cambia nel tempo, va osservata la contingenza (es. Taxi)
- Tardare equivale a non agire

■ Semi-dinamico

- L'ambiente non cambia ma la valutazione dell'agente sì
Esempio: Scacchi con timer





Discreto/continuo

- Possono assumere valori discreti o continui
 - lo stato: solo un numero finito di stati
 - il tempo
 - le percezioni
 - le azioni
- Focus prossime lezioni su stati discreti
- Ma la guida del taxi è un problema con stato e tempo continui ...
- Combinatoriale (nel discreto) versus infinito (nel continuo)



Noto/ignoto

- Distinzione riferita allo stato di conoscenza dell'agente sulle leggi fisiche dell'ambiente
- L'agente conosce l'ambiente oppure deve compiere azioni esplorative?
- Noto diverso da osservabile (es. carte coperte, ma regole note)

Ambienti reali: parzialmente osservabili, stocastici, sequenziali, dinamici, continui, multi-agente, ignoti



Tipologie di ambiente

Esercizio 2

	Osservabile /no	Deterministico/ stocastico	Episodico/s sequenziale	Statico/ dinamico	Discreto/ continuo	Mono/multi- agente?
Gioco 15	Osservabile	Deterministico	Sequenziale	Statico	Discreto	Mono
Briscola						
Scacchi						
Scacchi con tempo limitato						
Sudoku						
Taxi driver						



Simulatore di ambienti

Uno strumento software che si occupa di:

- generare stimoli per gli agenti
 - raccogliere le azioni in risposta
 - aggiornare lo stato dell'ambiente
 - [attivare altri processi che influenzano l'ambiente]
 - valutare le prestazioni degli agenti
-
- *Esperimenti* su classi di ambienti (variando le condizioni)
essenziale per valutare capacità di *generalizzare*
 - Valutazione prestazione come medie su più istanze

Sommario (cap. 2, ALMA)

- Agenti razionali
- Proprietà degli ambienti
- **La struttura degli agenti**
 - Agenti reattivi semplici
 - Agenti basati su modello
 - Agenti con obiettivo
 - Agenti con funzione di utilità
 - Agenti che apprendono

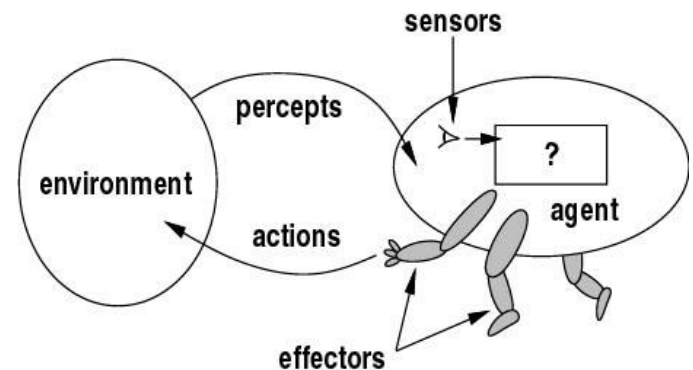
Rapida overview



Struttura di un agente

Agente = Architettura + Programma

Ag: P \rightarrow Az
percezioni *azioni*



Il programma dell'agente implementa la
funzione Ag

Programma agente

Molto generico

```
function Skeleton-Agent (percept) returns action
  static: memory, the agent's memory of the world
  memory ← UpdateMemory(memory, percept)
  action ← Choose-Best-Action(memory)
  memory ← UpdateMemory(memory, action)
  return action
```

Agente basato su tabella

- La scelta dell'azione è un accesso a una «tabella» che associa un'azione ad ogni possibile sequenza di percezioni.

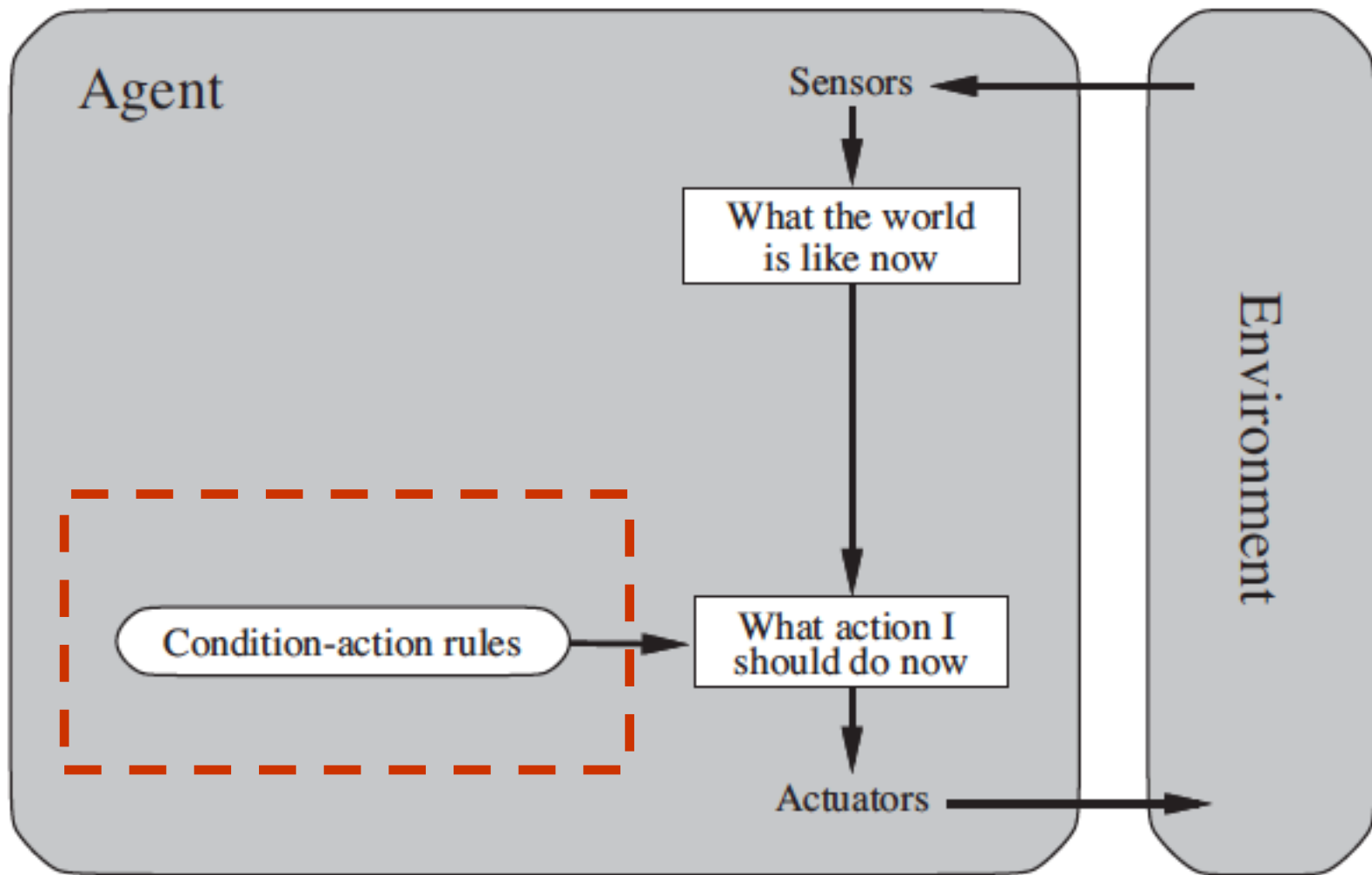
Problemi:

1. Dimensione: Per giocare a scacchi tabella con un numero di righe $\gg 10^{80}$ numero di atomi nell'universo!!! → ingestibile
2. Difficile da costruire
3. Nessuna autonomia
4. Di difficile aggiornamento, apprendimento complesso.

In IA vogliamo realizzare agenti razionali
con programma “*compatto*”

Agenti reattivi semplici

No storia in memoria

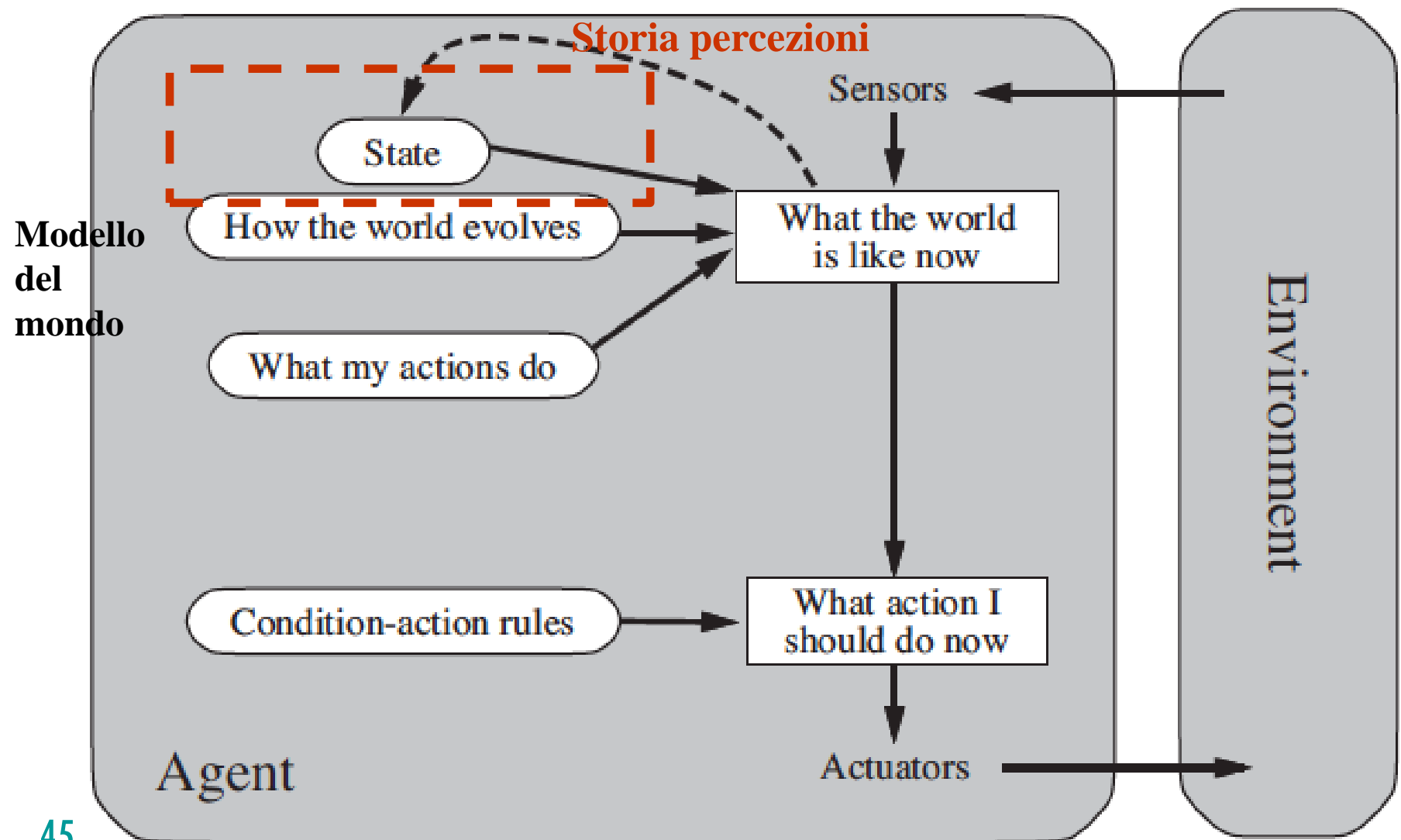




Agenti reattivi - programma

```
function Agente-Reattivo-Semplice (percezione)  
  returns azione  
  
  persistent: regole, un insieme di regole  
  condizione-azione (if-then)  
  
  stato  $\leftarrow$  Interpreta-Input(percezione)  
  regola  $\leftarrow$  Regola-Corrispondente(stato, regole)  
  azione  $\leftarrow$  regola.Azione  
  return azione
```

Agenti basati su modello





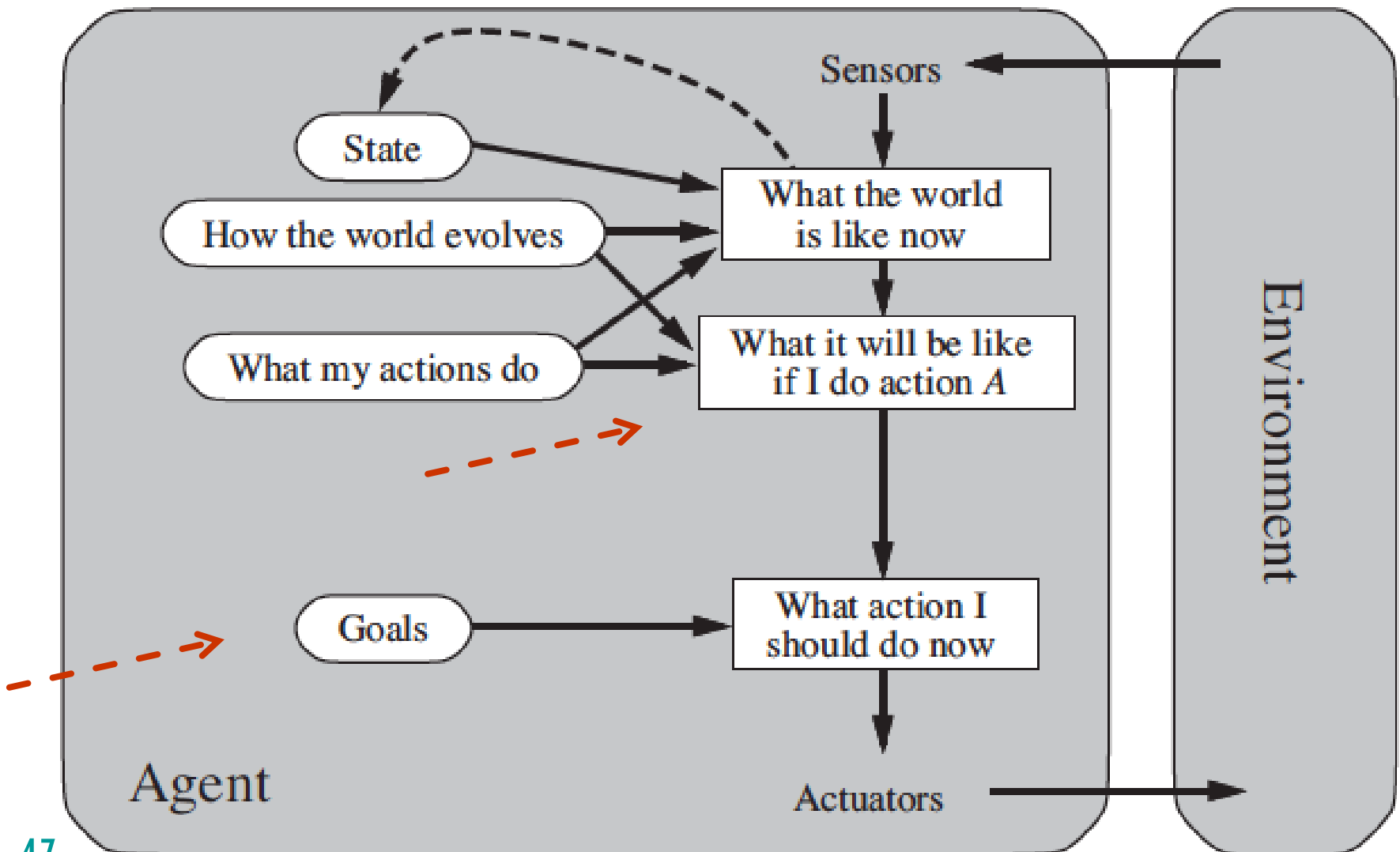
Agenti basati su modello

```
function Agente-Basato-su-Modello (percezione)  
  returns azione  
  persistent: stato, una descrizione dello stato corrente  
               modello, conoscenza del mondo  
               regole, un insieme di regole condizione-azione  
               azione, l'azione più recente  
  [stato ← Aggiorna-Stato(stato, azione, percez., modello) ]  
  regola ← Regola-Corrispondente(stato, regole)  
  azione ← regola.Azione  
return azione
```

Agenti con obiettivo

Ove bisogna pianificare una sequenza di azioni per raggiungere l'obiettivo (**goal**)

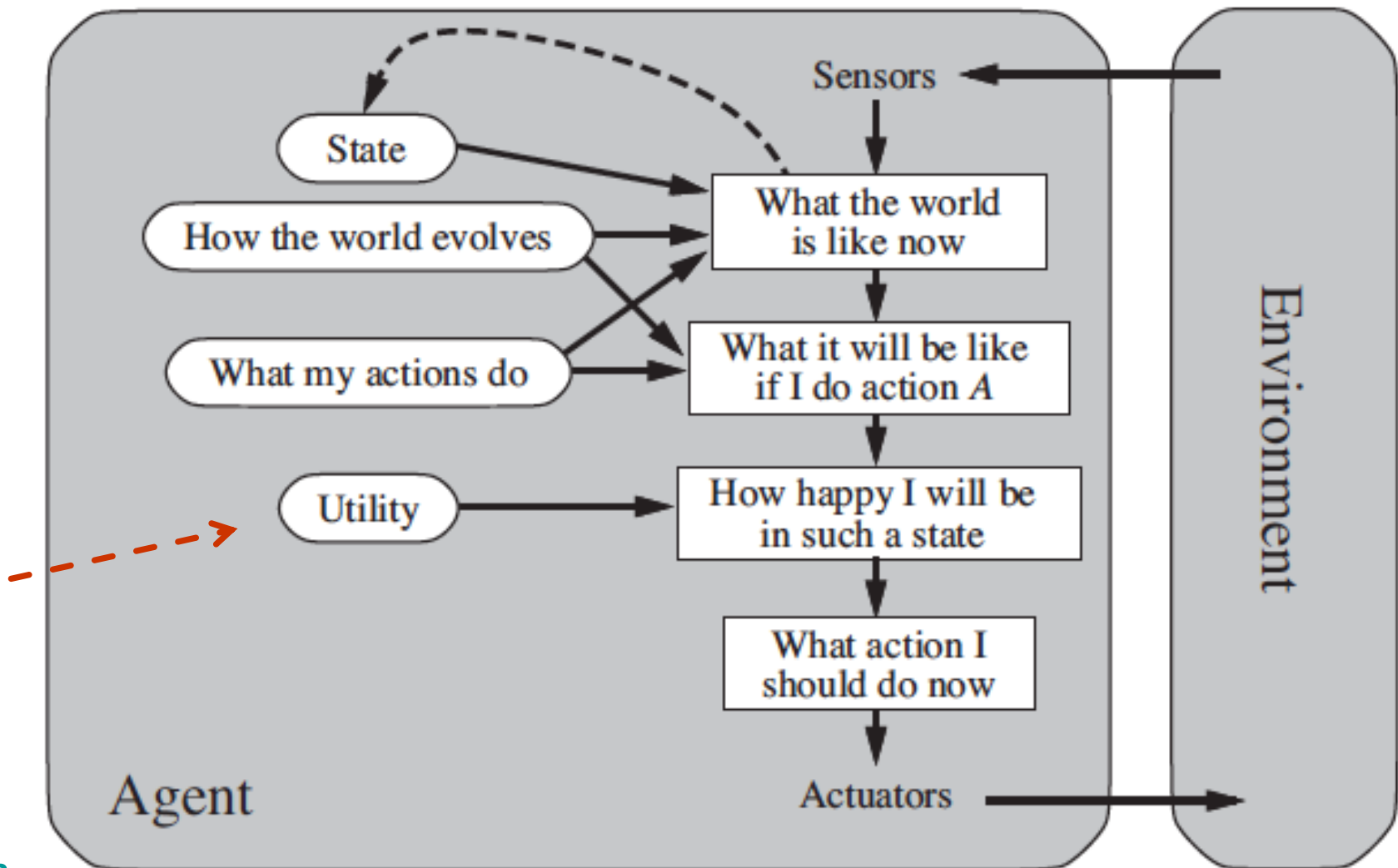
Parti aggiunte



Agenti con obiettivo

- Sono guidati da un obiettivo nella scelta dell'azione (è fornito un goal esplicito: e.g. Città da raggiungere)
 - A volte l'azione migliore dipende da qual è l'obiettivo da raggiungere (es. da che parte devo girare?).
 - Devono *pianificare una sequenza di azioni* per raggiungere l'obiettivo.
 - Meno efficienti ma più *flessibili* di un agente reattivo (obiettivo può cambiare, non è già codificato nelle regole)
 - Es. Ricerca di seq. di azioni (cammino) per una destinazione x

Agenti con valutazione di utilità

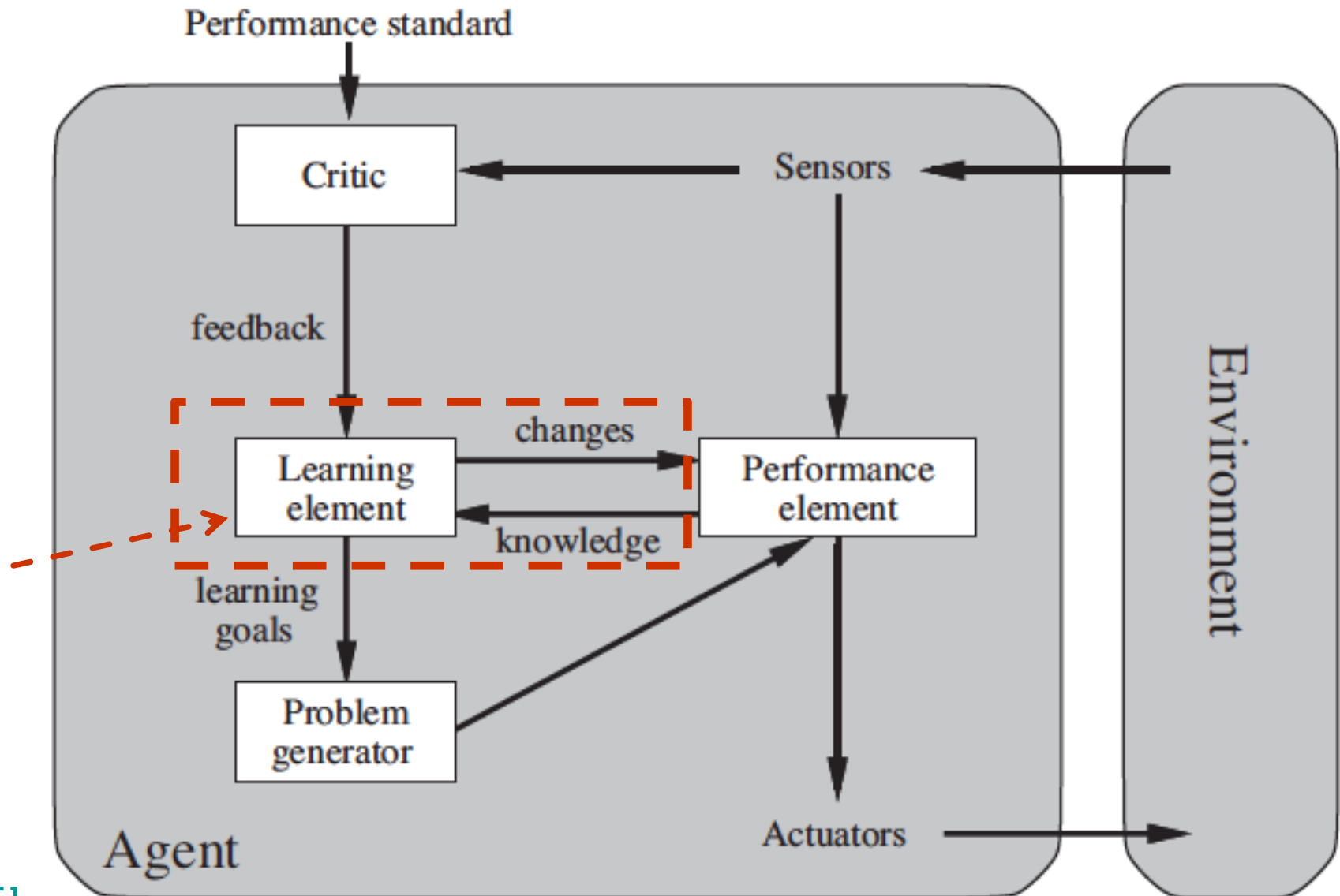




Agenti con valutazione di utilità

- Obiettivi alternativi (o più modi per raggiungerlo)
 - l'agente deve decidere verso quali di questi muoversi.
 - necessaria una funzione di **utilità** (che associa ad uno stato obiettivo un numero reale).
- Obiettivi più facilmente raggiungibili di altri
 - la funzione di utilità tiene conto anche della probabilità di successo (e/o di ciascun risultato): **utilità attesa** (o in media)

Agenti che apprendono



Agenti che apprendono

1. Componente di apprendimento

- Produce cambiamenti al programma agente
- Migliora prestazioni , adattando i suoi componenti, apprendendo dall'ambiente

2. Elemento esecutivo (“Performance element”)

- Il programma agente (visto sinora per decider le azioni)

3. Elemento critico

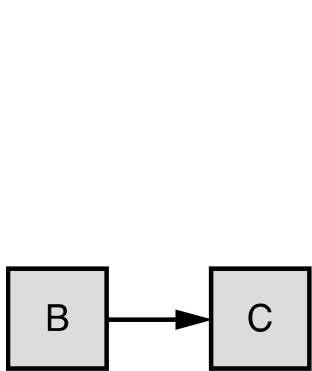
- Osserva e dà feedback sul comportamento

4. Generatore di problemi

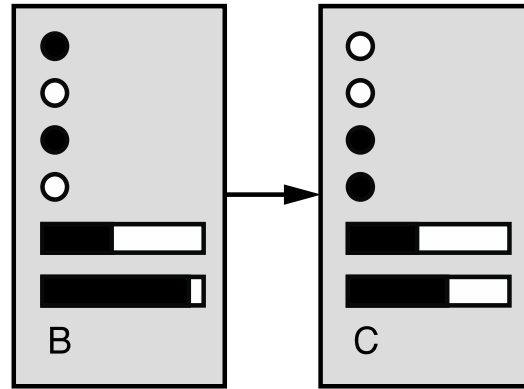
- Suggerisce nuove situazioni da esplorare

Tipi di rappresentazione

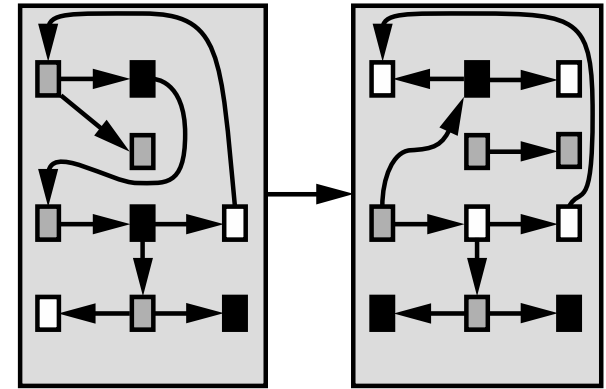
Stati e transizioni



(a) Atomic



(b) Factored



(b) Structured

- Rappresentazione atomica (stati)
- Rappresentazione fattorizzata (piu variabili e attributi)
- Rappresentazione strutturata (aggiunge relazioni)

Conclusioni

- Un po' astratti ma ...punto di vista unificante: IA come progettazione di agenti e programmi agente
 - Rivedetela poi come «collante» tra le varie parti del corso.
- Misure di prestazioni e razionalità (max mis. prestazione)
- Classificazione degli ambienti operativi
- Diversi schemi di complessità crescente per i programmi agente
 - Diverso tipo di inf esplicitata e utilizzata per decidere
 - Dipendenza dalla natura dell'ambiente
- Tutti gli agenti possono migliorarsi con l'apprendimento
- BIB (Bibliografia): AIMA Capitolo 2

Per informazioni

Alessio Micheli

micheli@di.unipi.it



**Dipartimento di Informatica
Università di Pisa - Italy**



**Computational Intelligence &
Machine Learning Group**