

Appunti di Sistemi Complessi: Modelli e Simulazioni

A cura di:
Francesco Refolli
Matricola 865955

Anno Accademico 2023-2024

Chapter 1

Introduzione

1.1 Classical AI

Agentificazione dell'AI Un singolo agente che risolve problemi, adottando varie soluzioni e tecniche. Ma cos'è un agente?

Secondo Russel e Norvig , un agente è qualsiasi cosa che percepisce il suo ambiente tramite dei sensori e agisce su esso tramite degli attuatori.

Secondo Wooldridge-Jennings un agente è un sistema computazionale che gode di queste proprietà: autonomia, abilità sociale, reattività e pro-attività.

Secondo Jacques Ferber un Agente è un'entità fisica o virtuale:

- Che è capace di agire in un ambiente
- Che può comunicare direttamente con altri agenti
- Che è guidato da un insieme di tendenze (sotto forma di obiettivi individuali o di una funzione di soddisfazione/sopravvivenza che cerca di ottimizzare)
- Che possiede risorse proprie
- Che è in grado di percepire il suo ambiente (ma in misura limitata)
- Che ha solo una rappresentazione parziale di questo ambiente (e forse nessuna)
- Che possiede competenze e può offrire servizi
- Il cui comportamento tende a soddisfare i propri obiettivi, tenendo conto delle risorse e delle competenze a sua disposizione e in funzione della sua percezione, della sua rappresentazione e delle comunicazioni che riceve (Jacques Ferber)

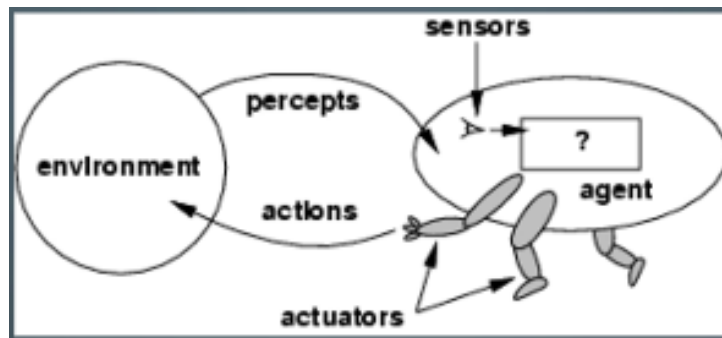


Figure 1.1: Schema

Agenti Umani : occhi, orecchie e altri organi sono sensori; mani, piedi ... etc sono gli attuatori.

Agenti Robotici : telecamere e scanner infrarossi sono sensori; varie pompe idrauliche o motori possono essere gli attuatori.

La agent function mappa lo storico delle percezioni alle azioni ($f : P^* \rightarrow A$).

L'agent program esegue sull'architettura fisica per produrre le azioni di f .

Agent = Architecture + Program

1.2 Distributed AI

Passiamo da un singolo agente intelligente che implementa una soluzione ad un sistema di entita' che nell'insieme risolve un problema interagendo con l'ambiente circostante. Ma cos'e' un sistema?

Un sistema puo' essere definito come un gruppo di elementi che interagiscono regolarmente o indipendenti che formano un tutt'uno (il sistema capitalista, un insieme di organi, un gruppo di devices ... etc). Dare una definizione di sistema comunque non e' facile.

Gia' definire cosa e' distribuito non e' poca cosa:

- Risoluzione distribuita di problemi
- Risoluzione di problemi distribuiti
- tecniche distribuite di risoluzione di problemi

Maes definisce gli agenti autonomi come sistemi computazionali che *abitano* un ambiente dinamico e complesso, sentono e agiscono in modo autonomo in questo ambiente per realizzare un set di goal o attivita' per cui sono stati creati.

Hayes-Roth definiscono gli agenti intelligenti come continuamente performanti tre funzioni: percezione di condizioni dinamiche dell'ambiente, azioni che hanno effetto su queste condizioni e ragionamenti per interpretare le percezioni, risolvere problemi, fare inferenze e determinare le azioni.

Chapter 2

Architettura di un Agente

2.1 Classificazione di Genesereth

Un agente puo' essere classificato in tre categorie secondo le loro "interiora" e le loro capacita'. Questa analisi era stata fatta considerando un singolo agente ma e' comunque valida con sistemi multi agente.

2.1.1 Tropistic

Questi agenti sono descrivibili come una tupla $\langle E, P, A, see, do, action \rangle$:

- E e' il set di stati dell'ambiente
- P e' una partizione di E che rappresenta una rilevante astrazione dell'ambiente dal punto di vista dell'agente
- A e' il set di azioni
- $see : E \rightarrow P$
- $action : P \rightarrow A$
- $do : A \times E \rightarrow E$

Questi agenti percepiscono l'ambiente, decidono un'azione e la attuano cambiando l'ambiente.



Figure 2.1: Tropistic Agent

2.1.2 Hysteretic

Questi agenti sono descrivibili come una tupla $\langle I, E, P, A, i_0, see, internal, do, action \rangle$:

- I e' il set di stati dell'agente
- E e' il set di stati dell'ambiente
- P e' una partizione di E che rappresenta una rilevante astrazione dell'ambiente dal punto di vista dell'agente
- A e' il set di azioni
- $see : E \rightarrow P$
- $action : I \times P \rightarrow A$
- $internal : I \times P \rightarrow I$
- $do : A \times E \rightarrow E$

Questi agenti percepiscono l'ambiente, decidono un'azione e la attuano cambiando l'ambiente e il loro stato interno.

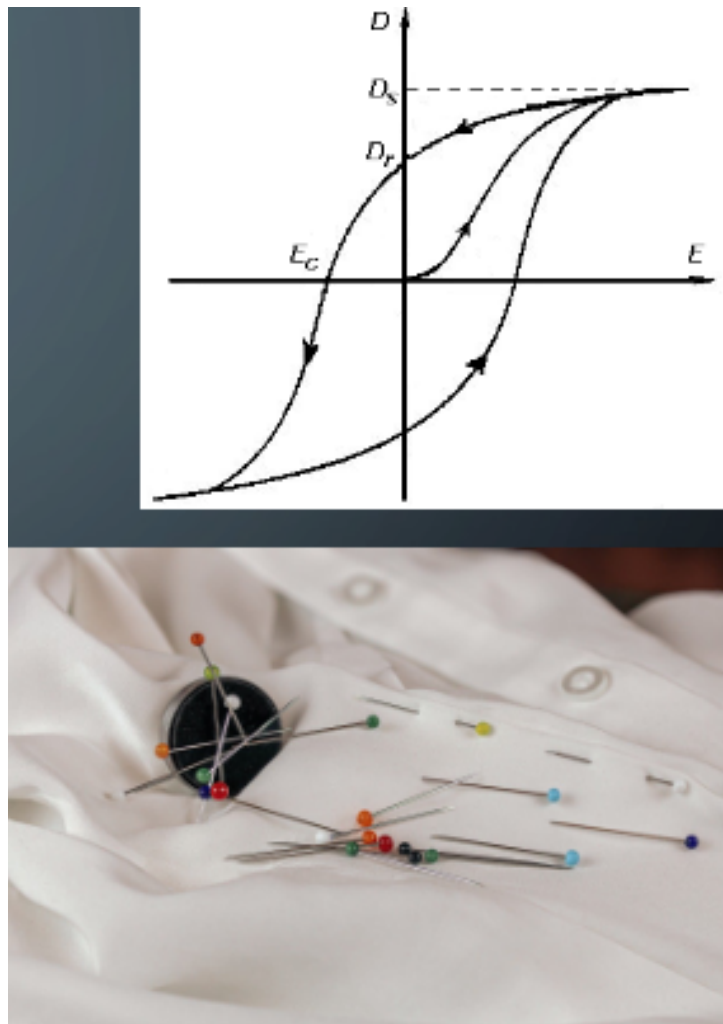


Figure 2.2: Hysteretic Agent

2.1.3 Knowledge-level

Questi agenti sono descrivibili come una tupla $\langle D, E, P, A, d_0, see, database, do, action \rangle$:

- D e' il set di predicati
- E e' il set di stati dell'ambiente
- P e' una partizione di E che rappresenta una rilevante astrazione dell'ambiente dal punto di vista dell'agente
- A e' il set di azioni
- $see : E \rightarrow P$
- $action : D \times P \rightarrow A$
- $database : D \times P \rightarrow D$
- $do : A \times E \rightarrow E$

Questi agenti percepiscono l'ambiente, decidono un'azione e la attuano cambiando l'ambiente secondo le loro conoscenze che vengono man mano aggiornate. Simile ai Hysteretic ma qui abbiamo regole espresse in forma logica per le conoscenze.

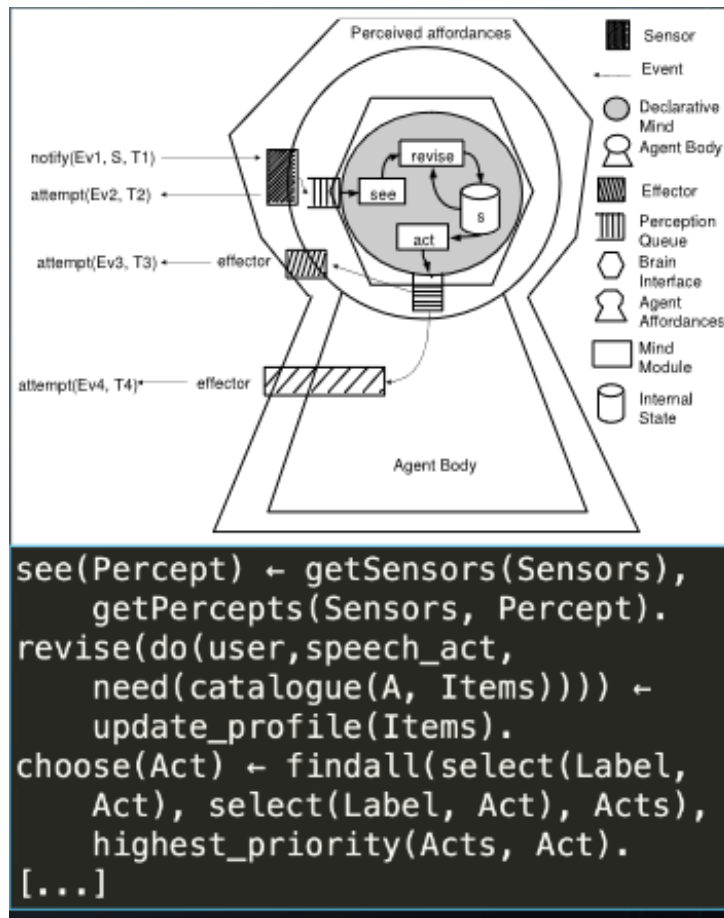


Figure 2.3: Knowledge-level Agent

2.2 Classificazione di Russel Norvig

Un agente può essere classificato in quattro categorie secondo la loro architettura interna.

2.2.1 Simple Reflex

Sono equivalenti agli agenti Tropistic.

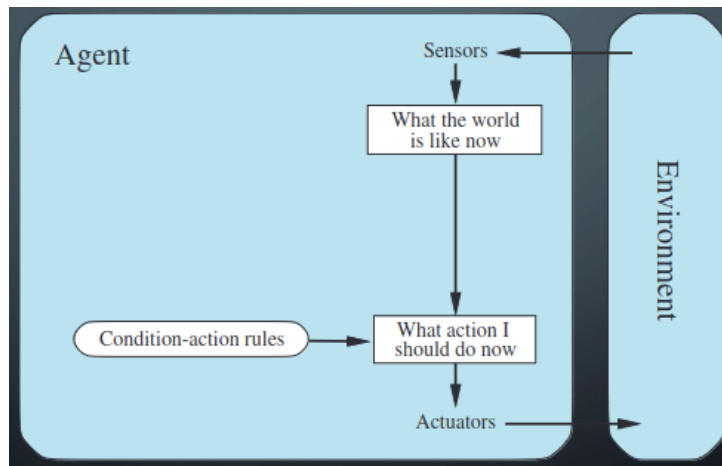


Figure 2.4: Simple Reflex Agent

2.2.2 Reflexive with Internal State

Sono equivalenti agli agenti Hysteretic.

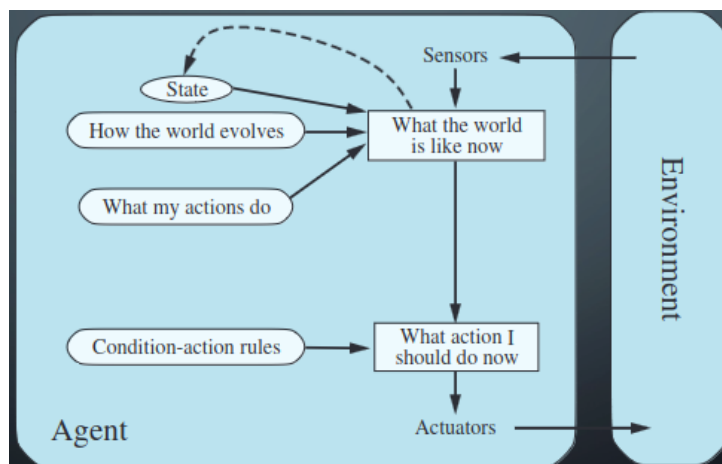


Figure 2.5: Reflexive Agent with Internal State

2.2.3 Goal-based

E' una specializzazione degli agenti Knowledge-level.

Questi agenti hanno bisogno di un goal per sapere quali situazioni sono desiderabili, e le cose diventano difficili con lunghe sequenze di azioni richieste per trovare un goal. La differenza fondamentale rispetto ad agenti piu' semplici e' la capacita' di prendere in considerazione in futuro in modo esplicito e formale. Sono flessibili perche' puo' manipolare la base di conoscenza essendo essa rappresentata esplicitamente.

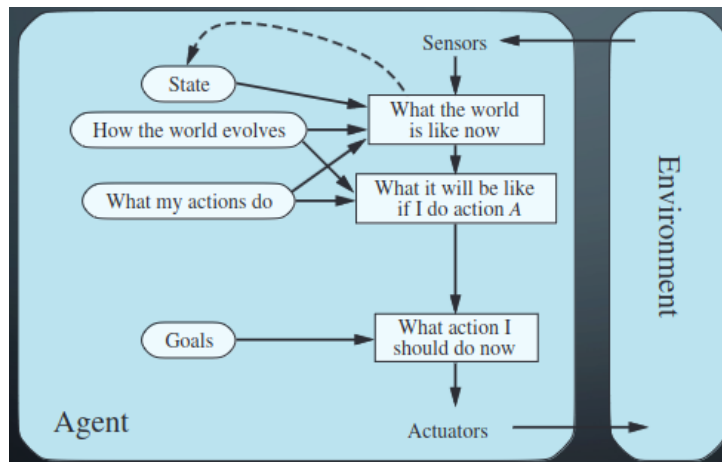


Figure 2.6: Goal-based Agent

2.2.4 Utility-based

E' una specializzazione degli agenti Knowledge-level. Alcuni goal possono essere raggiunti in modi diversi (alcuni piu' uguali di altri). La Utility e' una funzione che descrive una misura di performance rispetto a questi metodi. Seleziona i goal in base alla probabilita' di successo e risolve i conflitti tra goal.

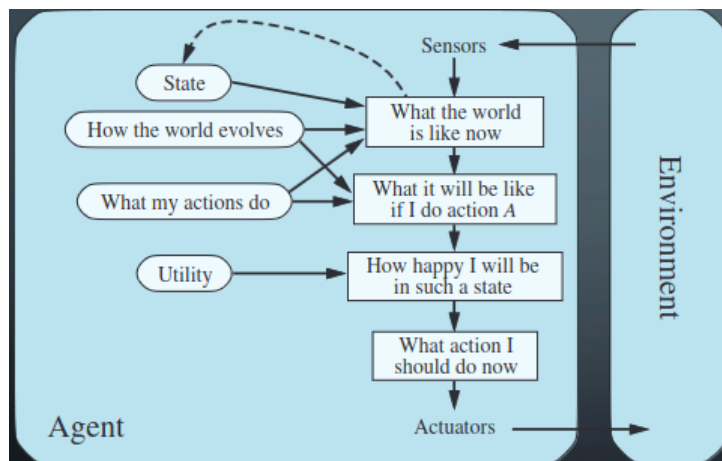


Figure 2.7: Utility-based Agent

2.3 Altre Classificazioni

Possono essere classificati in base alla capacita' (deliberativi vs reattivi, possibilita' di imparare, mobili vs statici (obsoleta)), oppure in base al ruolo nel sistema (Cooperativo, informativo, interfaccia). Chiaramente si possono creare sistemi eterogenei con agenti ibridi.

2.4 Esempi

2.4.1 Reattivi - I Boids

Un modello per coordinare il movimento di elementi in modo che agiscano come un fluido o come uno stormo. Fondamentalmente il loro comportamento non e' guidato da un goal (formalmente vanno solo avanti) ma la decisione di virare e' presa in base ad alcune regole (anche in contrasto tra loro, quindi pesate).

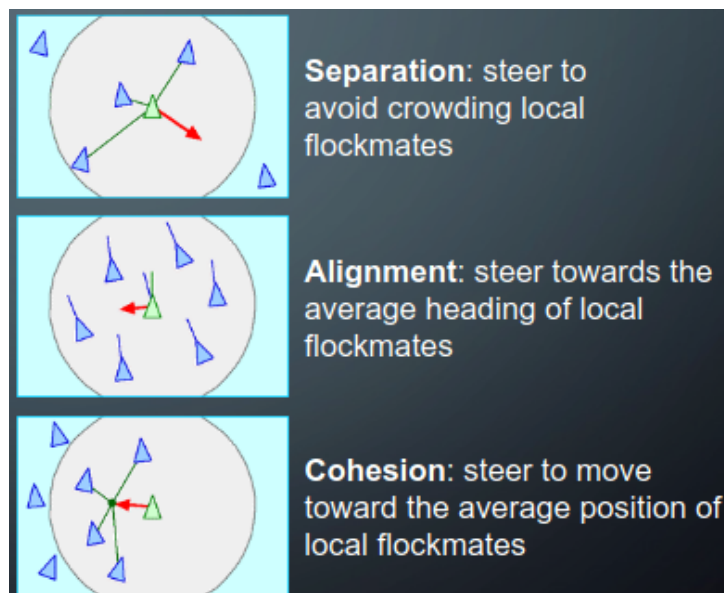


Figure 2.8: regole dei Boid

2.4.2 3APL

3APL e' un agente che utilizza una base di conoscenza e delle regole di derivazione logiche unite ad un goal globale per inferire le azioni da usare per risolvere problemi. Questo esempio prende in esame un ambiente statico, ma ci possono essere casi in cui l'ambiente e' modificabile o dinamico.

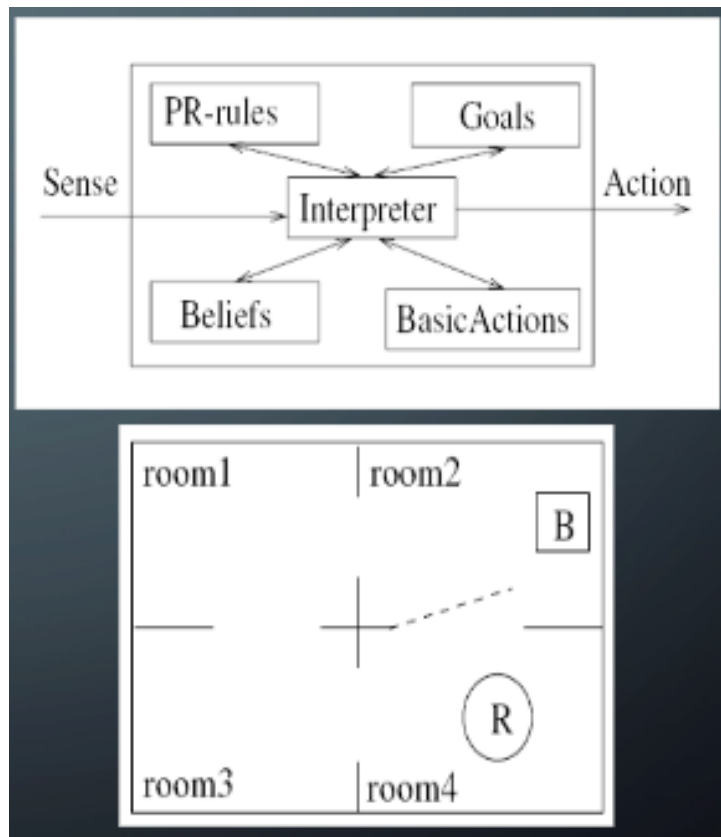


Figure 2.9: schema 3APL

2.5 JADE

JADE e' uno "strumento abilitante" (cioe' un software per eseguire agenti) sviluppato da Telecom. Non prescrive un'architettura precisa ma solo uno schema molto generale e minimale. Quindi resta al programmatore decidere come modellare i suoi agent. Dispone solo una gerarchia di classi per modellare il comportamento e un modello di esecuzione degli agenti.

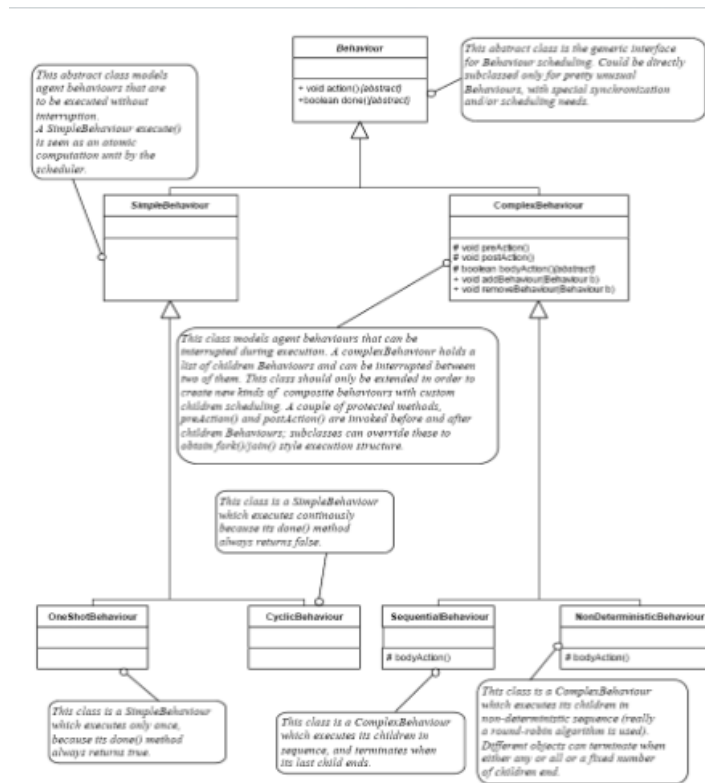


Figure 2.10: "Architettura" JADE

2.6 Affinita' Divergenze fra il Compagno Oggetto e Noi

Un agente e' un oggetto con autonomia, pro-attività e abilità sociale. Queste sono definizioni roboanti ma assolutamente labili, infatti non c'e' un'unica interpretazione di esse e quindi lasciano il tempo che trovano. Possiamo avere anche agenti ibridi in varie modalita', eccone due:

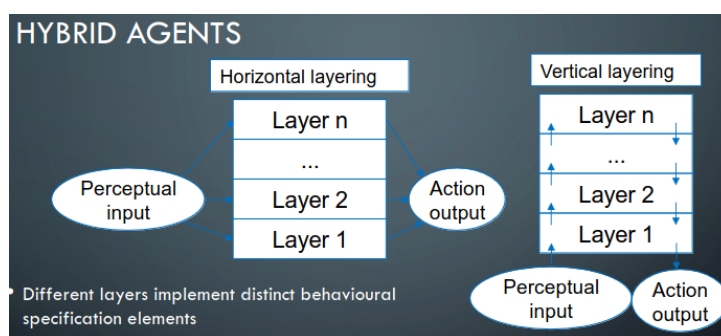


Figure 2.11: Agenti Ibridi

2.6.1 Autonomia

La parola autonomia ha diversi significati come detto prima:

- la capacita' di un agente di decidere le sue decisioni in termini di tempo, se rispondere una richiesta o senza uno stimolo esterno

- la capacita' di controllare il proprio stato interno
- la capacita' di prendere decisioni basate sulle proprie esperienze invece che su conoscenza hard-wired (= AGI?)

2.7 Agenti che Imparano

Tutti gli agenti precedenti utilizzavano descrizioni preconfezionate per decidere che cosa fare, tutta via c'è la possibilità di creare agenti che imparano da soli sotto certe condizioni cosa fare.

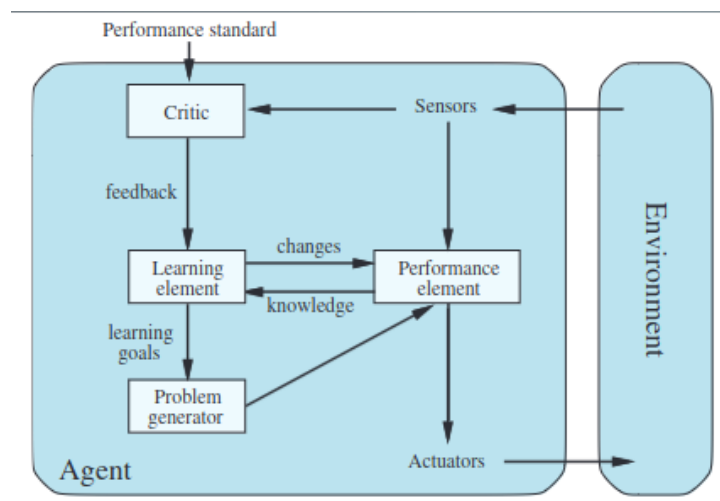


Figure 2.12: Learning Agent

2.7.1 Reinforcement Learning

Fondamentalmente un agente prende una decisione dato uno stimolo, e riceve un "dono" o una "penalita'" in base a se la decisione era buona o meno. Chiaramente si assume che tutti i goal possano essere rappresentati come la massimizzazione del reward cumulato.

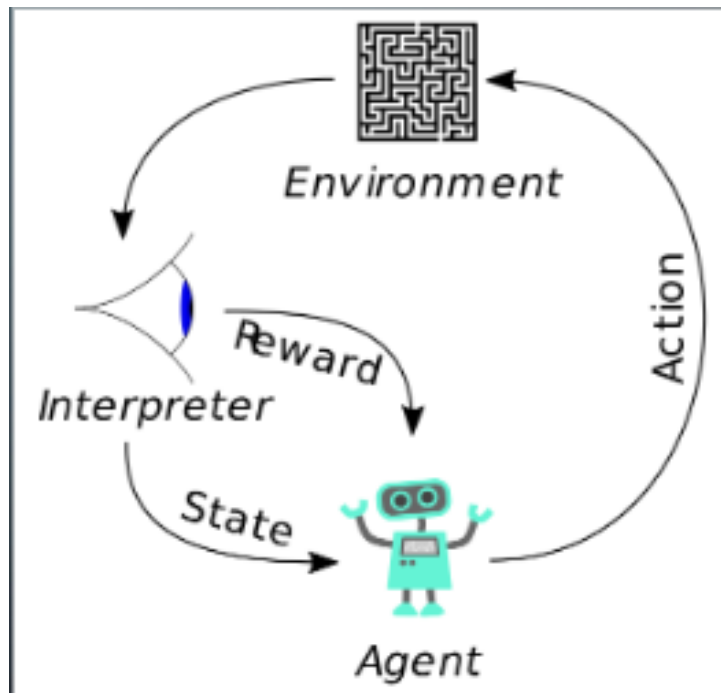


Figure 2.13: Schema

I problemi centrali sono di due nature:

- e' necessario formalizzare il tutto in termini matematici e algoritmici.
- l'agente deve forzatamente esplorare lo spazio degli stati per acquisire abbastanza esperienza da poter usare nel campo.