

# Indice

**Introduzione . . . . . 3**  
Agente intelligente . . . . . 3



# Capitolo 1

## Introduzione

### 1.1 Agente intelligente

Si definisce **agente intelligente**, o semplicemente **agente**, qualsiasi entità in grado di percepire l'ambiente in cui si trova mediante sensori e modificando tale ambiente compiendo delle azioni, mappando percezioni ad azioni. Con **ambiente** si intende la parte di universo a disposizione delle percezioni dell'agente e da questa influenzabile. L'intelligenza artificiale è definibile come lo studio degli agenti.

Un essere umano può essere modellato come un agente, potendo percepire l'ambiente tramite occhi, orecchie e altri organi e agendo su di esso per mezzo dei suoi arti. Allo stesso modo, un robot può essere modellato come un agente, percependo l'ambiente attraverso telecamere o sensori infrarossi e agendo su di esso mediante appendici e/o motori elettrici. Infine, anche un programma per computer può essere modellato come un agente, se si considera l'input umano (tramite tastiera, mouse, touchscreen o voce) come percezione ed il suo output (scrivere su un file, mostrare un contenuto a schermo, generare un suono, eccetera) come azione compiuta sull'ambiente.

La sequenza di percezioni di un agente è la storia completa di tutto ciò che l'agente ha percepito. In generale, la scelta dell'azione compiuta da un agente in un certo istante dipende dalla sua conoscenza a priori e/o dall'intera sequenza di percezioni precedente. Formalmente, il comportamento di un agente è descritto da una funzione agente che mappa sequenze di percezioni in azioni:  $f : \text{Pow}(P) \rightarrow A$ . Tale funzione è un concetto astratto, una caratterizzazione *esterna* di un agente: *internamente*, la funzione agente di un agente intelligente è implementata da un **programma agente**; tale funzione viene eseguita da un dispositivo elettronico dotato di sensori di sorta, chiamato **architettura**.

Un **agente razionale** è un agente che "fa la scelta giusta". La nozione di "scelta giusta" comunemente adottata nel campo dell'intelligenza artificiale è il **conseguenzialismo**: il comportamento dell'agente è valutato sulla base delle conseguenze delle sue azioni. Se un agente, in relazione ad una certa percezione, compie una azione desiderabile dal punto di vista dell'utilizzatore, allora tale agente ha compiuto la "scelta giusta", ed è definibile agente razionale. La nozione di desiderabilità viene descritta da una **misura di prestazione** che valuta ogni sequenza di stati in cui l'ambiente si trova. In genere, è preferibile definire una misura di prestazione rispetto a ciò che si vuole accada all'ambiente piuttosto che rispetto al modo in cui ci si aspetta che funzioni.

È allora possibile fornire una definizione operativa di agente razionale: per ogni possibile sequenza di percezioni, un agente razionale sceglierà di compiere l'azione che, sulla base delle percezioni precedenti e sulla base della conoscenza che possiede a priori, restituisce il massimo valore possibile in termini di misura di prestazione. Si noti come "razionale" non significhi "onnisciente", ovvero in grado di prevedere con assoluta certezza ciò che accadrà in futuro, dato che questo è realisticamente impossibile; un agente razionale deve limitarsi a compiere azioni che massimizzano la prestazione *attesa*.

La definizione di agente razionale sopra presentata prevede che questo possieda anche una qualche nozione di **apprendimento**: per quanto la sua configurazione iniziale possa essere fissata, questa può venire modificata e potenziata con l'esperienza. Nel caso in cui l'ambiente sia interamente conosciuto a priori, l'agente non ha alcuna forma di apprendimento, limitandosi a compiere le azioni preimpostate.

Un agente che compie azioni esclusivamente sulla base della sua conoscenza a priori e non fa uso di apprendimento si dice che non è **autonomo**. Un agente razionale dovrebbe invece essere autonomo, ovvero partire sì da una base di conoscenza pregressa ma, attraverso l'apprendimento, colmarne le lacune. Dopo abbastanza esperienza, ci si aspetta che un agente razionale diventi di fatto indipendente dalla sua conoscenza a priori. È possibile classificare gli ambienti rispetto a cinque metriche informali, utili a ragionare sulla difficoltà del problema e sulla modalità risolutiva da adottare:

- **Accessibile o inaccessibile.** Un ambiente è tanto accessibile quanto un agente è in grado di ottenere le informazioni sul suo stato di cui necessita con completa accuratezza. Un ambiente può essere inaccessibile perché i sensori dell'agente non sono precisi oppure perché parte dell'ambiente è del tutto preclusa ai sensori dell'agente. Gli ambienti nel mondo reale hanno necessariamente un certo grado di inaccessibilità;
- **Deterministico o non deterministico.** Un ambiente è deterministico (in riferimento alle azioni dell'agente) se la sua evoluzione è completamente determinata dal suo stato attuale e dalle azioni dell'agente. Un ambiente è non deterministico se la sua evoluzione è anche influenzata da forze al di là dell'agente. Il mondo fisico da modellare ha sempre un certo grado di non determinismo;
- **Episodico o sequenziale.** In un ambiente episodico l'esperienza di un agente può essere divisa in step atomici dove la scelta di un'azione dipende esclusivamente dalla percezione attuale. In un ambiente sequenziale le azioni che un agente compie possono dipendere del tutto o in parte da quali azioni sono state prese in precedenza;
- **Statico o dinamico.** Un ambiente è statico se non subisce modifiche mentre l'agente sta deliberando, altrimenti è dinamico;
- **Discreto o continuo.** Un ambiente è discreto se il numero di stati in cui questo può trovarsi è finito, ovvero se è possibile (almeno in linea teorica) enumerare tutti i suoi possibili stati, altrimenti è continuo. Essendo i computer discreti per definizione, modellare un ambiente continuo attraverso un sistema automatico richiederà sempre un certo grado di approssimazione.

- Si consideri come ambiente il gioco degli scacchi e come agenti i giocatori umani (si assuma che le mosse non abbiano alcun limite di tempo). Tale ambiente é:
  1. Accessibile, perché ciascun giocatore ha completa conoscenza dello stato della partita;
  2. Deterministico, perché l'evoluzione degli stati dipende esclusivamente da quali mosse scelgono di compiere i giocatori;
  3. Sequenziale, perché le mosse di un giocatore possono anche dipendere da quali mosse ha compiuto in precedenza;
  4. Statico, perché durante l'esecuzione di una mossa e durante la scelta della stessa lo stato della partita rimane invariato;
  5. Discreto, perché il numero di possibili stati in cui la partita può trovarsi é finito.
- Si consideri come ambiente le strade di una città e come agente un sistema di guida automatico per automobili. Tale ambiente é:
  1. Inaccessibile, perché non é possibile conoscere l'intero stato del traffico di tutta la città in ciascun istante;
  2. Non deterministico, perché l'evoluzione del traffico non dipende esclusivamente dalle scelte dell'agente;
  3. Sequenziale, perché la scelta di quale strada percorrere può dipendere anche da quali strade ha percorso in precedenza;
  4. Dinamico, perché lo stato della città e del traffico cambiano anche mentre l'agente é in movimento;
  5. Continuo, perché lo stato della città e del traffico si modificano costantemente.

Gli agenti intelligenti possono essere informalmente classificati in quattro categorie, di crescente ordine di complessità.

### 1.1.1 Agenti con riflessi semplici

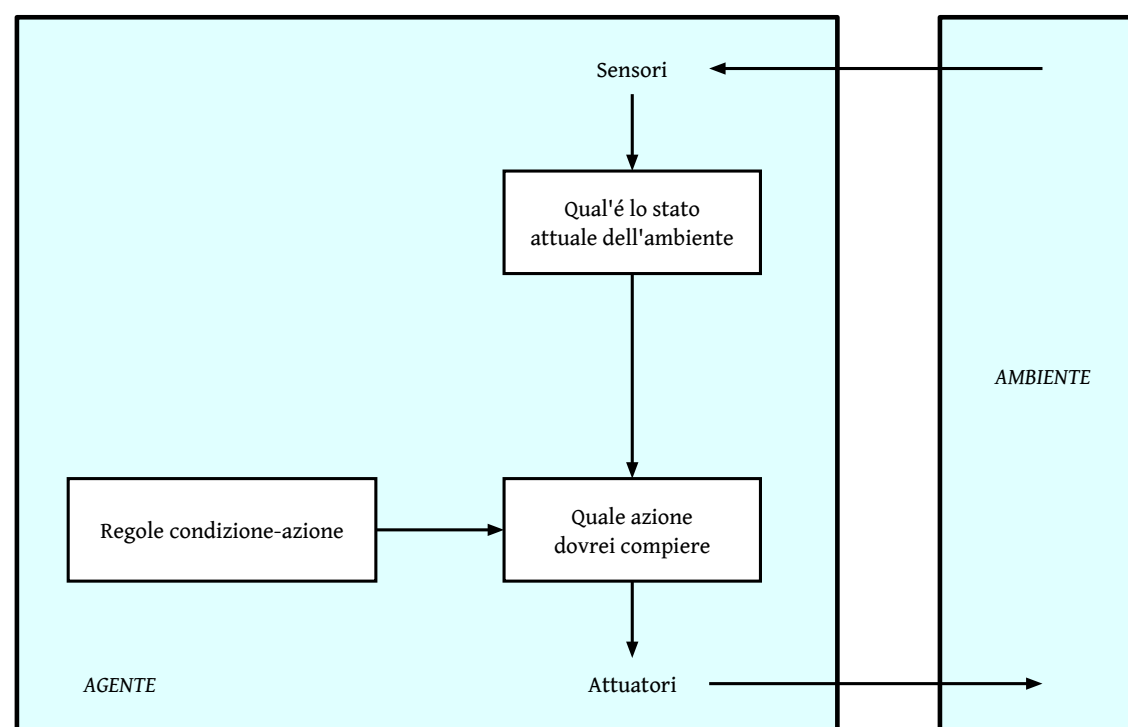
Gli agenti più facili da realizzare sono gli **agenti con riflessi semplici**. Questi agenti non hanno alcun modello dell'ambiente: scelgono che azione compiere esclusivamente sulla base della percezione attuale e non hanno cognizione delle percezioni precedenti.

Agenti di questo tipo scelgono che azioni compiere seguendo **regole condizione-azione**: se si verifica una certa condizione, allora viene compiuta l'azione associata a tale condizione.

Una rappresentazione schematica di un agente con riflessi semplici é presentata in basso. La funzione **INTERPRET-INPUT** genera una descrizione astratta della percezione ricevuta dall'agente, mentre la funzione **RULE-MATCH** restituisce la prima azione associata a tale rappresentazione di percezione nel set di regole **rules**.

```
rules <= set of condition-action rules
```

```
function SIMPLE-REFLEX-AGENT(percept)
state <= INTERPRETER-INPUT(percept)
rule <= RULE-MATCH(state, rules)
action <= rule.action
return action
```



Gli agenti con riflessi semplici hanno una intelligenza limitata. Infatti, agenti di questo tipo operano correttamente solamente se l'azione da compiere che massimizza la funzione di prestazione può essere determinata solo sulla base delle proprie percezioni, ovvero se l'ambiente é completamente accessibile. Se nella propria conoscenza a priori sono presenti errori o se l'ambiente é accessibile solo in parte, l'agente sarà destinato ad operare in maniera non razionale.

Ancora più problematica è la situazione in cui agenti con riflessi semplici entrano in loop infiniti, dato che non sono in grado di determinarli. L'unica contromisura che possono adottare è randomizzare le proprie azioni, dato che in questo modo si riduce la probabilità che l'agente compia le stesse azioni più volte di fila. Tuttavia, sebbene questo approccio possa mettere una pezza al problema del loop infinito in maniera semplice, in genere comporta uno spreco di risorse, e pertanto risulta difficilmente in un comportamento razionale da parte dell'agente.

1.1.2 Agenti con riflessi, ma basati su un modello

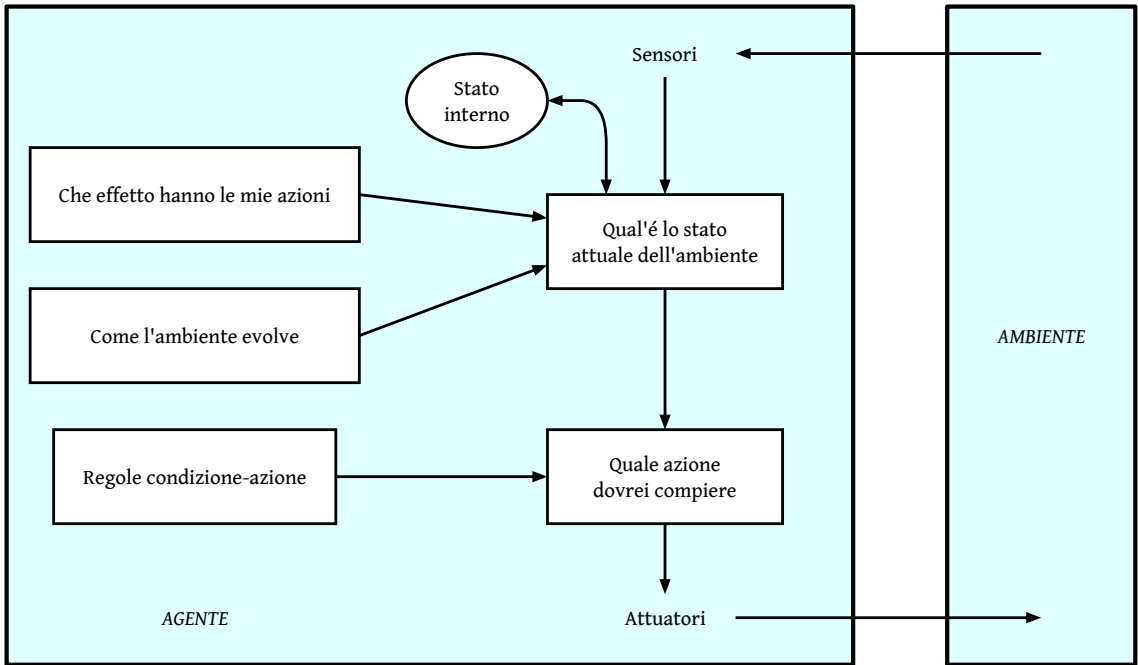
Il modo più efficiente per risolvere il problema dell'avere a che fare con un agente parzialmente accessibile è tenere traccia della parte di ambiente di cui questo non ha conoscenza. Ovvero, l'agente dovrebbe avere una qualche sorta di **stato interno** che dipende dalle percezioni che questo ha captato in precedenza, di modo da avere informazioni su alcuni degli stati diversi da quello corrente. Agenti di questo tipo sono detti **agenti con riflessi ma basati su un modello**.

Aggiornare periodicamente tale stato interno richiede che l'agente possieda due forme di conoscenza. Innanzitutto, è necessario avere informazioni relative al modo in cui l'ambiente si evolve nel tempo, sia in termini di come le azioni dell'agente influenzano l'ambiente che in termini di come l'ambiente si evolve in maniera indipendente dall'agente. Questo corpo di informazioni prende il nome di **modello di transizione**. Inoltre, è necessario avere informazioni relative a come l'evoluzione dell'ambiente si riflette sulle percezioni dell'agente, nel complesso chiamate **modello sensoriale**.

Una rappresentazione schematica di un agente con riflessi ma basati su un modello è presentata in basso, dove la funzione UPDATE-STATE aggiorna lo stato interno dell'agente prima di restituire l'azione da compiere.

```
state <= the agent's current conception of the environment state
transition_model <= a description on how the next state depends on the current state and action
sensor_model <= a description on how the current world state is reflected in the agent's percepts
rules <= set of condition-action rules
action <= the most recent action (starts NULL)

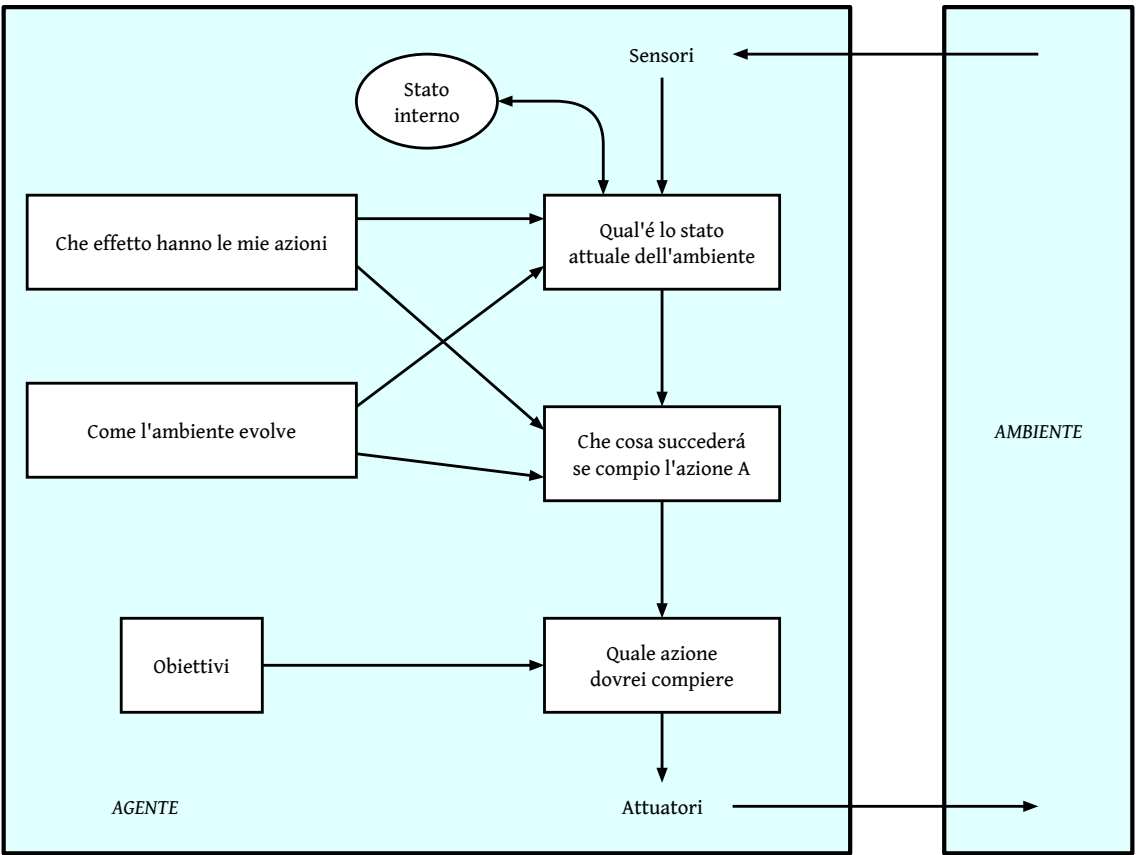
function MODEL-BASED-REFLEX-AGENT(percept)
state <= UPDATE-STATE(state, action, percept, transition_model, sensor_model)
rule <= RULE-MATCH(state, rules)
action <= rule.action
return action
```



Si noti come difficilmente un agente con riflesso basato su un modello può determinare con certezza lo stato attuale dell'ambiente. In genere, un agente può limitarsi ad averne una descrizione parziale.

1.1.3 Agenti basati su un modello, ma basati su obiettivi

Vi sono situazioni in cui la scelta di quale sia l'azione migliore da compiere da parte di un agente dipenda anche da un qualche tipo di obiettivo a lungo termine. Non sempre questo obiettivo viene raggiunto nell'operare una sola azione, ma può richiedere diverse azioni intermedie. In agenti di questo tipo, la medesima azione ed il medesimo stato interno possono risultare in azioni diverse se è diverso l'obiettivo.

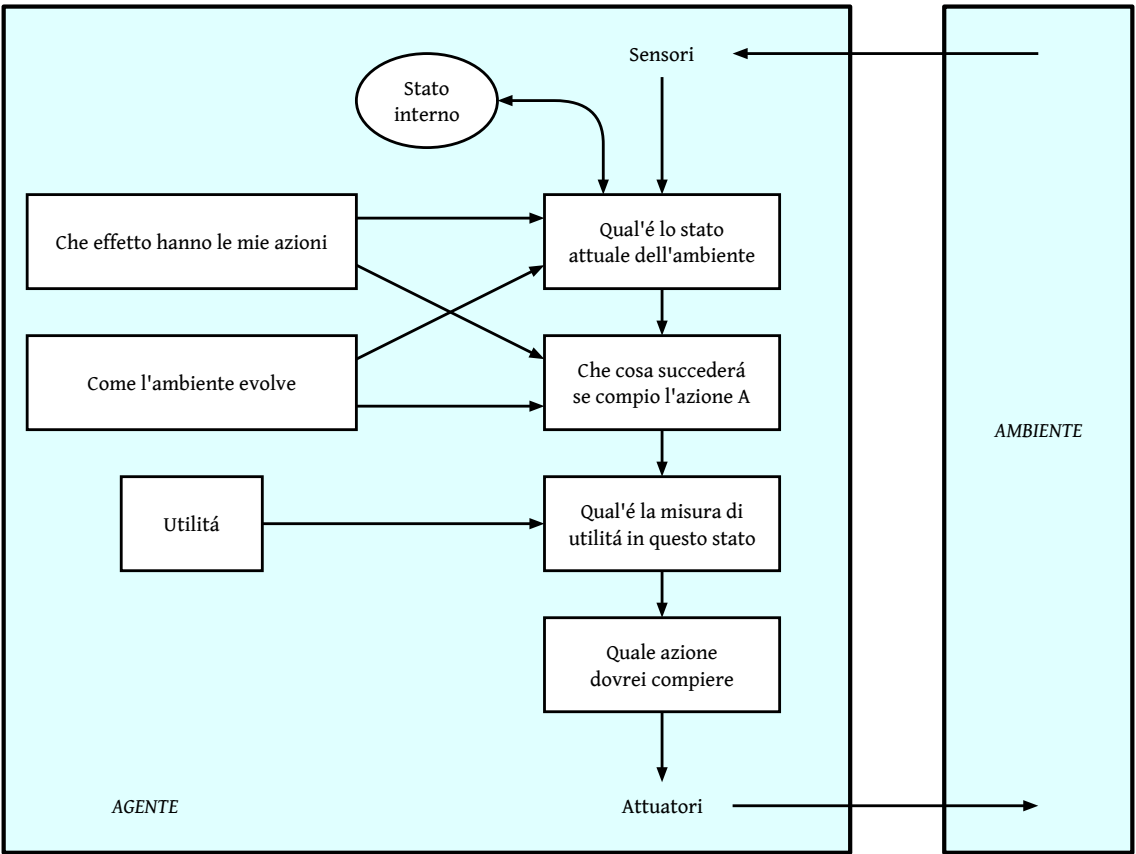


1.1.4 Agenti basati su un modello e guidati da utilità

Non sempre é possibile costruire un agente razionale semplicemente spingendolo a raggiungere un obiettivo. Infatti, se tale obiettivo può essere raggiunto tramite diverse sequenze di azioni, una potrebbe essere preferibile ad un'altra. Inoltre, un agente potrebbe dover perseguire più obiettivi contemporaneamente fra di loro incompatibili, ovvero compiere azioni che lo "avvicinano" ad un obiettivo ma al contempo "allontanarlo" da un altro.

Un obiettivo permette di discriminare gli stati dell'ambiente esclusivamente come "favorevoli" e "sfavorevoli", senza alcuna sfumatura nel mezzo. Un migliore approccio prevede invece di introdurre una misura di **utilità**, che influenza la scelta dell'agente nello scegliere quale azione compiere (insieme alla misura di prestazione, all'obiettivo da seguire e dal proprio stato interno).

La misura di utilità permette all'agente di, nel dover perseguire più obiettivi fra di loro incompatibili, scegliere l'azione che comporta il miglior compromesso nell'avanzamento di tutti loro. Inoltre, non sempre la struttura dell'ambiente garantisce che sia possibile raggiungere con assoluta certezza un obiettivo semplicemente eseguendo le azioni appropriate; anche in questo caso, la misura di utilità permette di valutare quanto sia "conveniente" per l'agente compiere una certa azione in vista di un determinato obiettivo sulla base di quanto sia ragionevole che tale obiettivo venga effettivamente raggiunto.



### 1.1.5 Agenti che apprendono

Gli agenti più interessanti sono indubbiamente quelli in grado di **apprendere**; tutti i tipi di agenti presentati finora possono essere costruiti come agenti che apprendono. Il notevole vantaggio che presentano è che possono operare in un ambiente del tutto sconosciuto apprendendo da questo, di modo da compiere le azioni migliori anche in situazioni dove lo stesso designer non ha modo di poter prevedere quali queste possano essere.

Un agente in grado di apprendere può essere concettualmente suddiviso in quattro componenti:

- La **componente di apprendimento**, che si occupa di migliorare la performance dell'agente;
- La **componente di performance**, che sceglie quale azione compiere sulla base delle percezioni e dello stato di conoscenza interno. Di fatto, questa componente costituiva l'intero agente dei modelli precedenti;
- Il **critico**, che informa la componente di apprendimento di quanto l'agente si sta comportando in maniera ottimale (razionale) sulla base di uno standard di performance prestabilito. Questa componente è necessaria perché le percezioni, di per loro, non sono in grado di informare l'agente sull'ottimalità del proprio comportamento;
- Il **generatore di problemi**, che suggerisce azioni all'agente che possono comportare nuove ed informative esperienze. Questa componente è necessaria perché se l'agente si affidasse esclusivamente alla componente di performance sceglierebbe sempre le azioni migliori sulla base della sua conoscenza attuale, che non sono necessariamente complete. Il generatore di problemi può portare l'agente a compiere azioni che possono potenzialmente essere localmente subottimali ma che sul lungo termine possono portare a compiere azioni ancora migliori.

