


L'intelligenza biologica


Sommario

Il neurone, modelli deterministici (L-system) e stocastici (frattali).
Reti Neurali.
Apprendimento con Rinforzo (Reinforcement Learning).
Mappe topologiche e clustering.
RBF: reti neurali con neuroni a base radiale.

La corteccia

Evoluzione storica - I

albori

- •1943 Warren McCulloch (neurofisiologo) & Walter Pitts (matematico)
  - •Modello di neurone elementare a soglia
- •1949 Donald Hebb
  - •Teorie sull'apprendimento
- •1960 Widrow & Hoff
  - •Delta rule; Adaline
- •1961 Steinbuck
  - •Memorie associative
- •1961 Caianiello
  - •Teoria statistica
- •1962 Rosenblatt
  - •Perceptrone; perceptron learning rule
- •1969 Minsky & Papert
  - •Problemi di apprendimento del perceptrone

periodo "romantico"

Evolzione storica - II

separazione del connessionismo dall'intelligenza artificiale simbolica

- •1968 Anderson
  - •Memorie associative
- •1974 Kohonen
  - •Memorie associative, mappe autoorganizzanti
- •1983 Barto, Sutton and Anderson
  - •Reinforcement Learning
- •1983 Hinton e Sejnoswky
  - •Unità stocastiche
- •1985 Amit
  - •Spin glass
- •1985 Rumelhart, Hinton & Parker
  - •Back propagation
- •1974 Werbos (economista)
  - •Back propagation
- •1989 Kohonen
  - •Memorie associative, mappe autoorganizzanti
- •1998 Vapnik
  - •Teoria dell'apprendimento e Support Vector Machines per problemi di classificazione

"revival"

Reinforcement learning

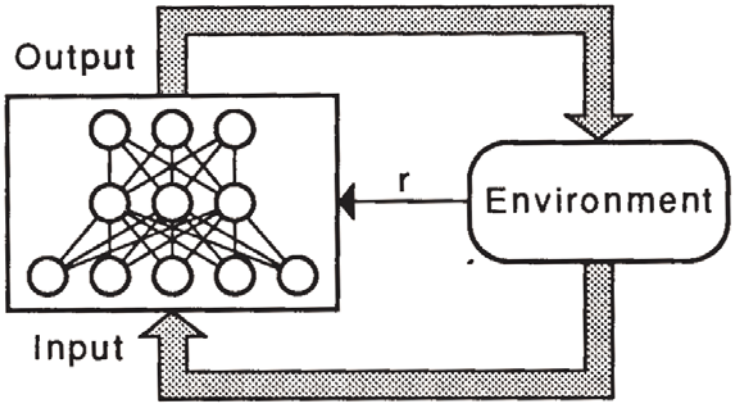
Nell'apprendimento supervisionato, esiste un "teacher" che dice al sistema quale è l'uscita corretta (learning with a teacher). Non sempre e' possibile.

Spesso si ha a disposizione solamente un'informazione giusto/sbagliato successo/fallimento.

Questa è un'informazione qualitativa ➔ ***learning with a critic.***

L'informazione disponibile si chiama ***segnale di rinforzo.*** *Non dà alcuna informazione su come aggiornare i pesi. Non è possibile definire una funzione costo o un gradiente.*

Reinforcement learning



Rete: Funzione non-lineare multi-input / multi-output.
Ambiente: scalare, r.

Condizionamento classico

"learning is an adaptive change of behavior and that is indeed the reason of its existence in animals and man (K. Lorentz, 1977).

Condizionamento classico. La risposta riflessa ad uno stimolo incondizionato viene evocata da uno stimolo condizionato.

Esperimenti di Pavlov. Campanello (stimolo condizionante), cibo (stimolo), risposta (salivazione).

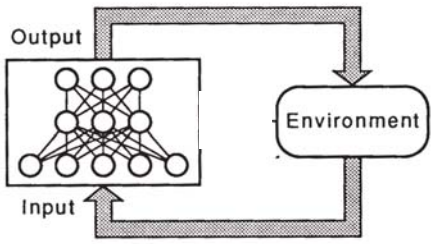
Stimolo-Risposta. Lo stimolo condizionante triggera una risposta condizionata.

Cf. Apprendimento Hebbiano.

Condizionamento operante

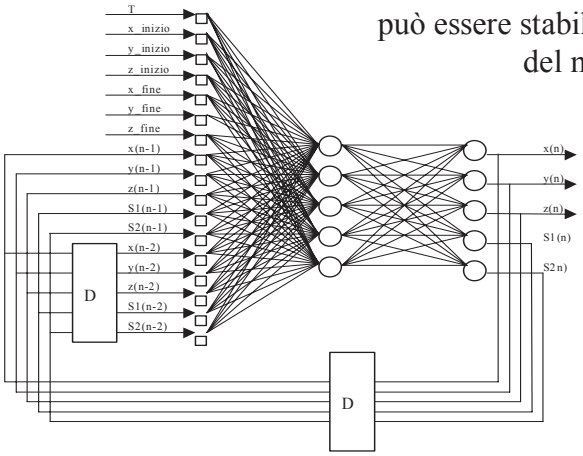
Condizionamento operante (reinforcement learning).

Interessa un comportamento. Una catena di input / output che può essere modificata agendo sul sistema. Il condizionamento agisce a ritroso sul sistema.



Esempio di condizionamento operante

Generazione di traiettorie, la correttezza può essere stabilita solamente alla fine del movimento.



Macchina di Huffman

Altro esempio: gioco degli scacchi.

Aspetti comuni dell'apprendimento

Stimolo. Input.

Risposta. Output.

Variazione della relazione input/output. Aggiornamento dei pesi.

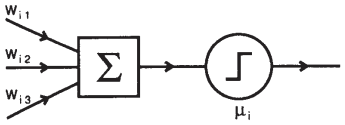
La variazione è attivata dallo stimolo condizionante. Come trasformare uno stimolo eterogeneo rispetto alla risposta in uno stimolo efficace?

Apprendimento con rinforzo di pattern di input/output

Classe I. Nel caso più semplice, il segnale di rinforzo è disponibile per ogni coppia di segnali ingresso/uscita. Esiste cioè una trasformazione definita tra ingresso e uscita che la rete deve imparare.

Questa è simile alla situazione di apprendimento supervisionato.
Rosenblatt **perceptron learning rule (neurone binario a soglia)**:

$$\Delta w_{ij} = \eta \Theta\left(y_i^D h_i\right) y_i^D u_j$$



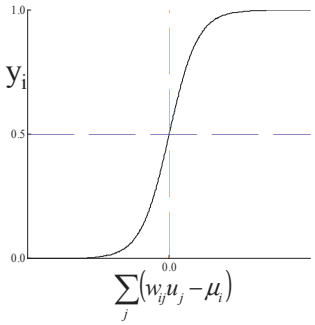
$\Theta(\bullet) \Rightarrow (y_i^D h_i) \Rightarrow y_i^D$, decide solo se la correzione deve essere effettuata.
y_i^D può essere interpretato come yes/no.

Apprendimento con rinforzo di pattern di input/output – funzioni di attivazione non-lineari

$$J = E(\mathbf{w}) = \frac{1}{2}\sum_p \left[\sum_i \left(y_{ip}^D - y_{ip}\right)^2 = \frac{1}{2}\sum_i \left(y_{ip}^D - (\sum_j w_{ij} u_{jp}) \right)^2 \right]$$

$$\Delta w_{ijp} = +\eta\left(y_i^D - y_i\right) g'\left(\sum_j w_{ij} u_j\right) u_j$$

Le condizioni:
$y_{ip} > y_{ip}^D$ e $y_{ip} < y_{ip}^D$
sono considerate allo stesso modo nel segnale di rinforzo (sbagliate).



Apprendimento con rinforzo in ambienti stocastici

Classe II. Questo tipo è generalmente applicato ad ambienti stocastici. In questo caso una particolare coppia ingresso/uscita determina una certa ***probabilità*** che il rinforzo sia positivo. La probabilità è comunque fissata (stazionaria) per ogni coppia ingresso/uscita.

Esempio two-armed bandit problem.

Massimizzare il reward, minimizzando il rischio.

Stochastic learning automata.

Trade-off tra exploration ed exploitation.

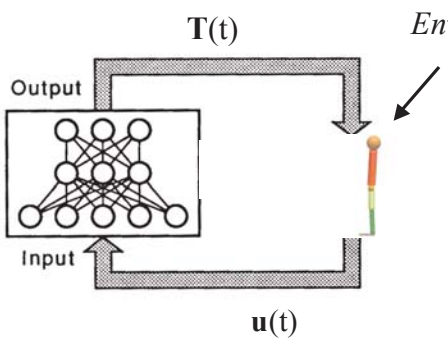
Apprendimento con rinforzo del comportamento di sistemi dinamici

Classe III. Nel caso più generale l'ambiente stesso è governato da leggi dinamiche molto complesse. Sia il segnale di rinforzo che gli ingressi dipendono dalla storia passata delle uscite della rete.

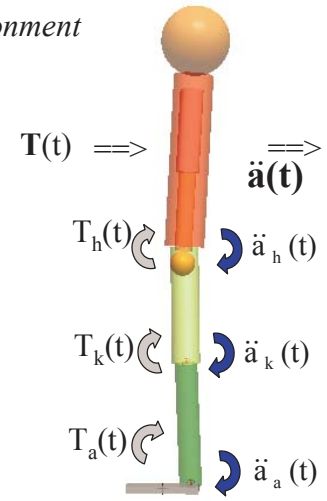
L'applicazione più classica è quella del gioco, dove l'ambiente rappresenta l'altro giocatore o gli altri giocatori. Se si considera per esempio il gioco degli scacchi, il segnale di rinforzo (vittoria o sconfitta) è inviato alle rete solo dopo un numero elevato di mosse. Applicazioni simili sono state sviluppate anche in psicologia dinamica.

Più recentemente un numero sempre crescente di applicazioni sono state sviluppate nell'ambito del controllo di sistemi complessi in ambienti non noti.

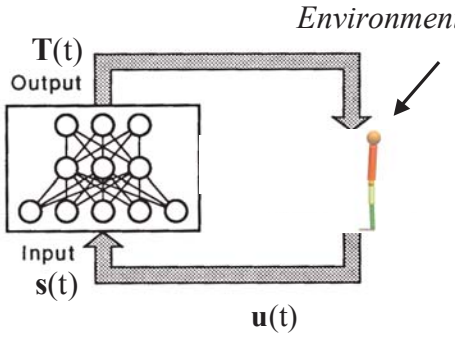
Apprendimento del controllo della postura di un robot umanoide.



$\ddot{\mathbf{a}}(\mathbf{t}) \rightarrow \mathbf{u}(t) = [\dot{\mathbf{a}}(t); \mathbf{a}(t)]$



Comportamento iniziale



$\ddot{\mathbf{a}}(\mathbf{t}) \rightarrow \mathbf{s}(t) = [\dot{\mathbf{a}}(t); \mathbf{a}(t)]$

Credit Assignement

Temporal credit assignement. In che istante la rete ha sbagliato?

Structural credit assignement. Quale unità della rete ha sbagliato?

Riassunto

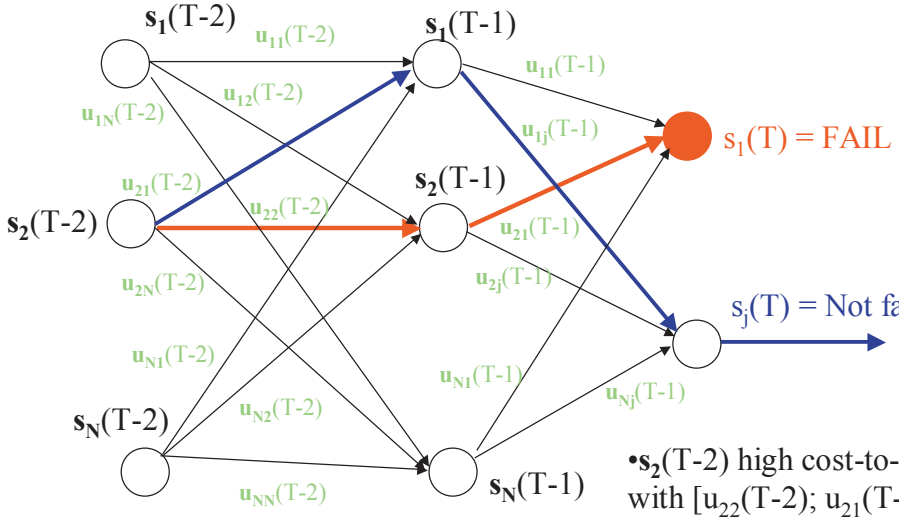
•Reinforcement learning. I pesi vengono modificati, rinforzando le soluzioni buone.

•Self-discovery of successful strategy. (it does not need to be optimal!). La strategia (di movimento, di gioco) non è data a-priori ma viene appresa attraverso trial-and-error.

•Credit assignement.

•Come possiamo procedere in modo efficiente nello scoprire una strategia di successo? Esplorazione dello spazio dei pesi?

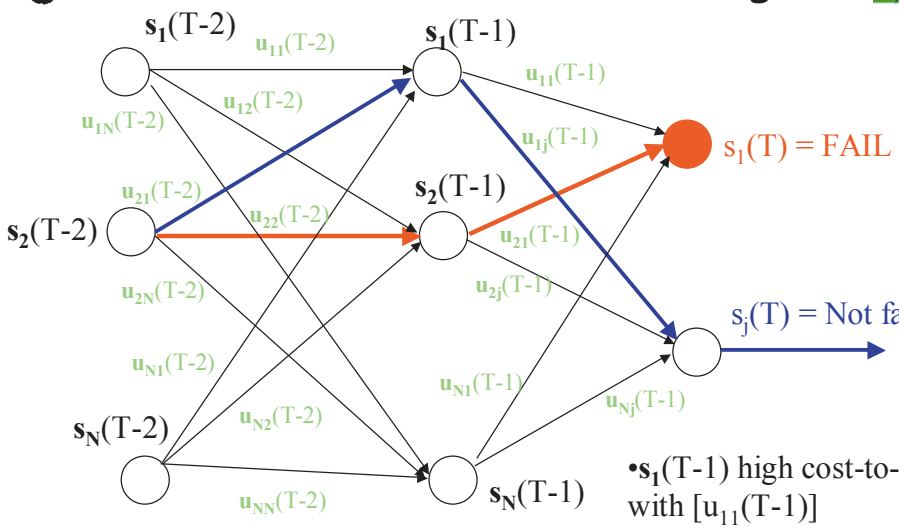
Il cost-to-go J(s(t))



Cost-to-go: {u(T-i)} + Cost of failure.

- $s_2(T-2)$ high cost-to-go with $[u_{22}(T-2); u_{21}(T-1)]$
- $s_2(T-2)$ low cost-to-go with $[u_{21}(T-2); u_{1j}(T-1)]$

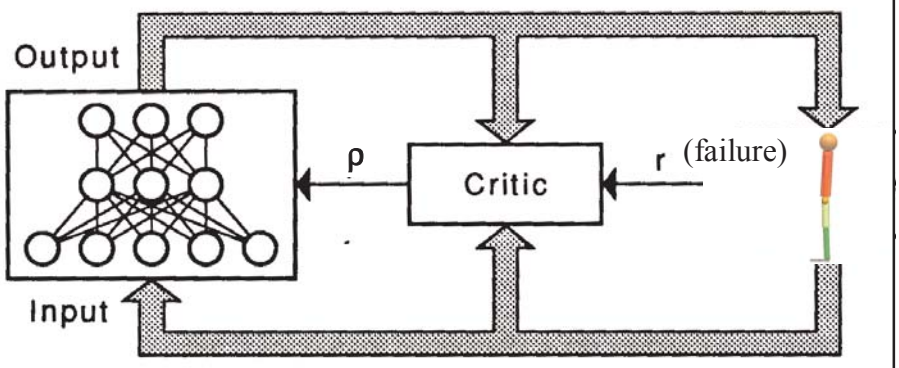
Come si determina il cost-to-go?



Partendo dallo stato finale, backwards.

- $s_1(T-1)$ high cost-to-go with $[u_{11}(T-1)]$
- $s_1(T-1)$ low cost-to-go with $[u_{ij}(T-1)]$

Reinforcement Learning



r is the primary reinforcement (failure), scalare.
ρ is the secondary reinforcement (derivato dal cost-to-go), scalare fornito con continuità nel tempo.

Problemi con la critica

La critica deve valutare il funzionamento del controllore in un modo che sia: **appropriato** per l'obbiettivo del controllo e sufficientemente **informativo** perché il controllore apprenda.

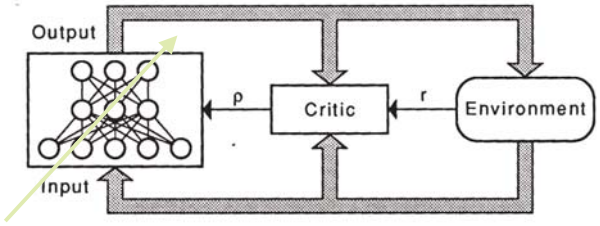
Determinare **come variare i pesi** del controllore in modo da migliorare le prestazioni, misurate dalla critica.

Come si utilizza la critica

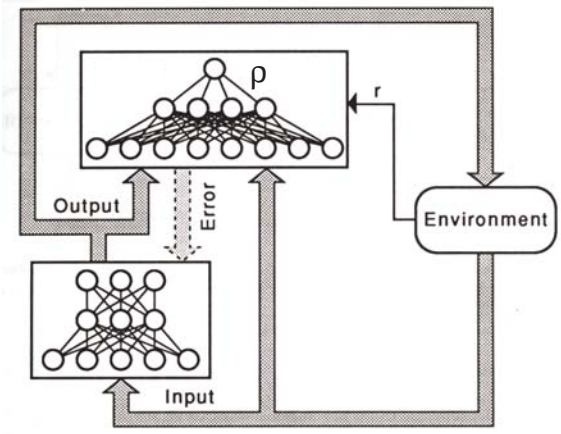
Per ogni istante t, il cost-to-go, J(t) = J(**s**(t)), viene rappresentato da una funzione non-lineare, derivabile.

E' possibile quindi calcolare il gradiente $dJ/ds|_t$ e determinare il nuovo stato: **s'**(t) = **s**(t) + d**s**(t) che migliora J(t): J(t)' = J(t) + dJ(t)

A sua volta, possiamo modificare i pesi del nostro controllore in modo tale che all'istante t, possiamo passare da s(t-1) a s'(t) = s(t) + ds(t).



Where does the cost-to-go come from?



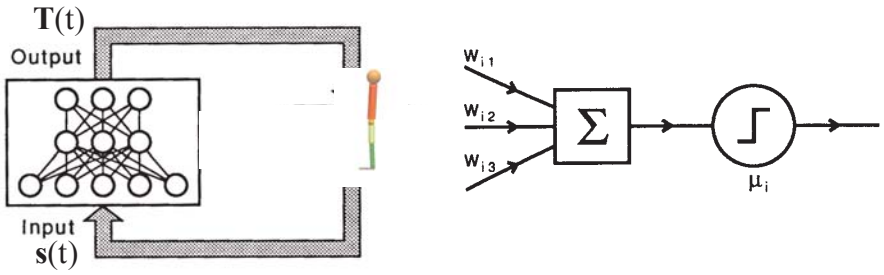
•Deve essere appreso anch'esso.

•Deve trasformare lo scalare r in un secondo salare ρ, fornito con continuità nel tempo.

•**Seconda rete neurale specializzata nell'apprendimento del cost-to-go.**

Il controllore e l'apprendimento

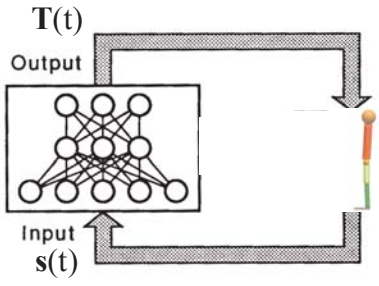
$$T_i(t) = \Theta\left(\sum_i w_{ij}(t) s_i(t) + \text{noise}(t)\right)$$



Noise(t) – ha il ruolo di incoraggiare l'esplorazione dello spazio.

L'eleggibillità

$$T_j(t) = \Theta\left(\sum_i w_{ij}(t) s_i(t) + \text{noise}(t)\right)$$



$$\Delta w_{ij}^{\ c} = \alpha \rho(t) e_{ij}(t)$$

$e_{ij}(t)$ – eleggibilità del peso ij.

Nel caso del perceptrone era:

$$\Delta w_{ij} = \eta \Theta\left(T_i^D T_i\right) T_i^D s_j$$

Il rinforzo, decide l'intensità dell'aggiornamento.
L'aggiornamento Hebbiano qui dipende dall'eleggibilità.

L'eleggibilità

$$e_{ij}^{c}(t+1) = \delta e_{ij}^{c}(t) + (1-\delta)T_j(t)s_i(t) \qquad \delta < 1$$

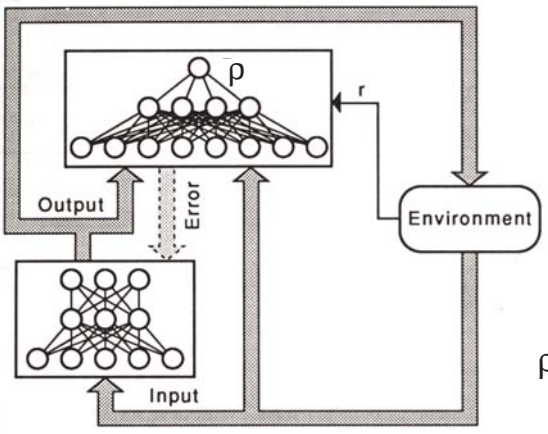
Se uno stato $s_i(t)$ non viene visitato ($s_i(t) = 0$), la sua eleggibilità decresce esponenzialmente.

Se uno stato $s_i(t)$ viene visitato di recente ($s_i(t) = 1$):
se $T_j(t)$ rimane dello stesso segno, la sua eleggibilità tende a $T_j * s_i$.
se $T_j(t)$ cambia spesso segno, la sua eleggibilità tende a 0.

La eleggibilità *aggiunge perciò la dimensione temporale al prodotto $T_j * s_i$: questo viene considerato valido solamente se si ripete nel tempo e se si ripete uguale.*

Il rinforzo interno ρ

Viene calcolato in due passi:



Viene innanzitutto calcolato per ogni istante di tempo, lo stato di rischio del sistema:

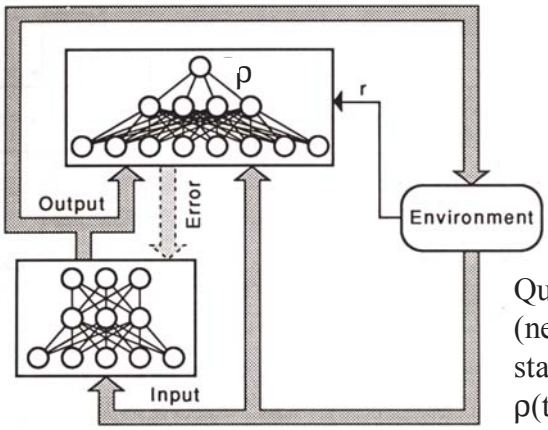
$$p(t) = \Theta\left(\sum_i w_i^{r}(t)s_i(t)\right)$$

e da p(t) il rinforzo interno:

$$\rho(t) = r(t) + \gamma p(t) - p(t-1)$$

Funzionamento del rinforzo interno

$$\rho(t) = r(t) + \gamma p(t) - p(t-1)$$



Fino a quando il controllore riesce a mantenere la postura eretta (nessun fallimento, r = 0), ρ(t) è **positivo**, quando il sistema passa da uno stato a più alto grado di rischio ad uno con un grado di rischio inferiore.

Quando arriva il reinforcement (negativo), r = -1. Non ci sono stati associati, per cui p(T) = 0. ρ(t) diventa **negativo**:
ρ(t) = –1 + p(t-1).

Apprendimento della mappa di rischio

$$\Delta w_i^r = \beta e_i^r(t) r(t)$$

$$e_i^r(t+1) = \lambda e_i^r(t) + (1-\lambda) s_i(t)$$

In questo caso l'eleggibilità riguarda solamente lo stato. E' sottointeso uscita(t) = r(t) = 1.

I modelli neurali del controllore e della critica

Le variabili sono codificate a **box**.

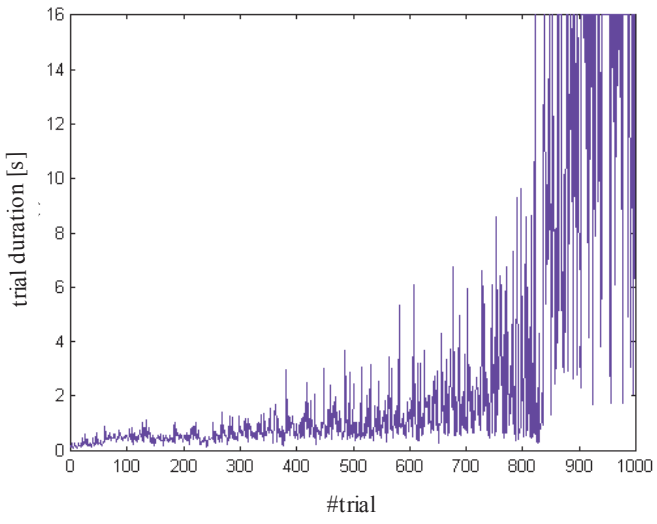
Orientamento del polpaccio rispetto ad un asse verticale $\vartheta : 0,\pm4,\pm12,\pm24$ deg
Velocità angolare del polpaccio $\dot{\vartheta} : \pm50,\pm\infty$ deg/s

Orientamento della coscia rispetto ad un asse verticale $\omega : 0,\pm4,\pm12,\pm24$ deg
Velocità angolare della coscia $\dot{\omega} : \pm50,\pm\infty$ deg/s

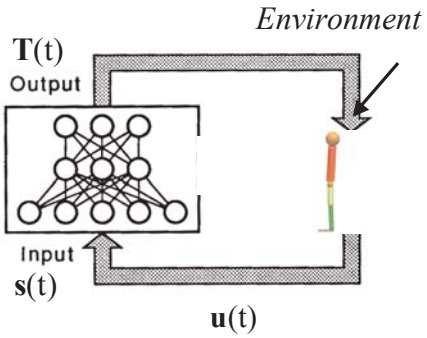
Orientamento del tronco rispetto ad un asse verticale $\varphi : 0,\pm4,\pm12,\pm24$ deg
Velocità angolare del tronco $\dot{\varphi} : \pm50,\pm\infty$ deg/s

Altra possibilità: fuzzy set. CMAC.

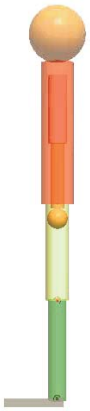
Curva di apprendimento



Apprendimento



$\ddot{a}(t) \rightarrow s(t)=[\dot{a}(t);a(t)]$



Riassunto sull'apprendimento con rinforzo

Necessita di una *critica*, che trasforma il segnale scalare di rinforzo (puntuale) in un segnale scalare temporale, r(T) -> ρ(t).

La critica analizza le coppie input/output ed impara una mappa di rischio.

Utilizza questa mappa di rischio per fornire un segnale di rinforzo interno al controllore.

Il controllore aggiorna i pesi con un meccanismo Hebbiano, dove il prodotto ingresso/uscita viene valutato lungo la dimensione temporale.