Relazione per il corso di
Sistemi Autonomi

Studio di un Distributed Adaptive Controller per Agenti Robotici: Reverse-Engineering del comportamento appreso

---

A.A.2017/2018

Alessandro Cevoli

Indice

1 Introduzione

Il breve studio esposto nelle prossime pagine si pone l'obiettivo di comprendere, nella maniera più intuitiva possibile, quali sono i pattern di attivazione neurale che portano l'agente realizzato a presentare un certo comportamento e le cause che portano allo sviluppo di tali sequenze di attivazione.

Nello specifico questo studio nasce sulla base di un precedente progetto [1] nella quale è stato riprodotto un Distributed Adaptive Controller [4]. Tale architettura permette ad un agente robotico di apprendere autonomamente uno o più comportamenti. In questo contesto si è realizzato il solo comportamento di Obstacle-Avoidance.

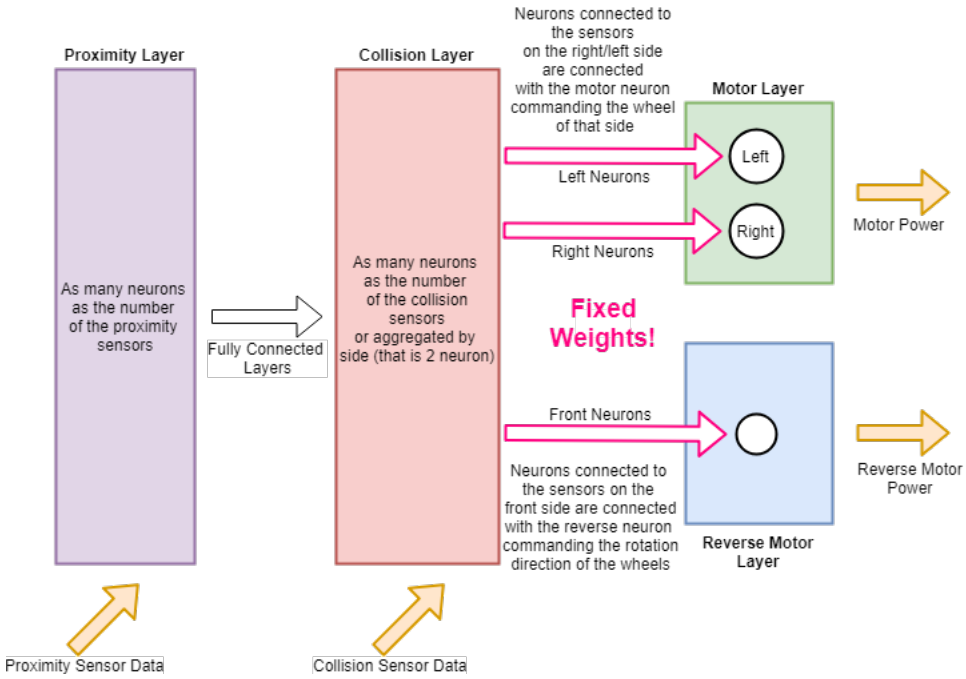


Figura 1: ANN Architecture

L'apprendimento, nel concreto, è implementato adoperando una rete neurale artificiale piuttosto semplice. Tale rete neurale presenta 2 layer di input, uno per i sensori di prossimità e un secondo per i sensori di collisione, totalmente connessi tra loro in maniera feed-forward (Proximity to Collision) e 2 layer di output. Un layer controlla la potenza erogata al motore delle ruote mentre l'altro controlla l'inversione di marcia di una delle 2 ruote. Come meccanismo di guida dell'apprendimento si è adoperato l'Hebbian Learning accoppiato ad un meccanismo di Active Forgetting, al fine di limitare la saturazione dei pesi della rete.

L'idea alla base del modello del Distributed Adaptive Controller è quello del Condizionamento Classico. All'agente vengono forniti uno stimolo condizionato (CS),

identificato nei valori percepiti dai sensori di prossimità, e uno stimolo non condizionato (US), identificato nei valori percepiti dai sensori di collisione. Tali feedback portano l'agente ad eseguire una o più risposte incondizionate (UR), identificate nelle azioni associate ad ognuno dei suoi neuroni del Motor Layer e del Reverse Layer. Dal momento che le connessioni tra il layer di neuroni che percepiscono gli stimoli non condizionati e i layer di neuroni che implementano le risposte sono definite durante la progettazione, quello che l'agente deve quindi apprendere è la corretta associazione $\{CS_i, US_j\}$, rappresentata dai pesi dello strato di connessioni feed-forward tra i due layer di input.

Il comportamento finale atteso è, dal punto di vista delle traiettorie e dei pattern di attivazione, simile a quello che l'agente dimostrerebbe se questi collidesse semplicemente con gli ostacoli e vi reagisse di conseguenza. L'obiettivo dell'apprendimento è quello di trovare la giusta configurazione di pesi della rete tale che permetta all'agente di anticipare il reale impatto con l'ostacolo, anticipando i pattern di attivazione.

Come presentato nel corso di [1], il robot (visibile in Figura 2) ha dimostrato alcuni comportamenti inaspettati che, dal nostro punto di vista (pilotato per lo più dal "buon senso"), appaiono "errati" o poco intelligenti. Tale demarcazione tra "corretto" ed "errato" diviene però sottile e a tratti sfumata dal momento che l'agente apprende il comportamento in maniera totalmente autonoma, secondo il suo "punto di vista".

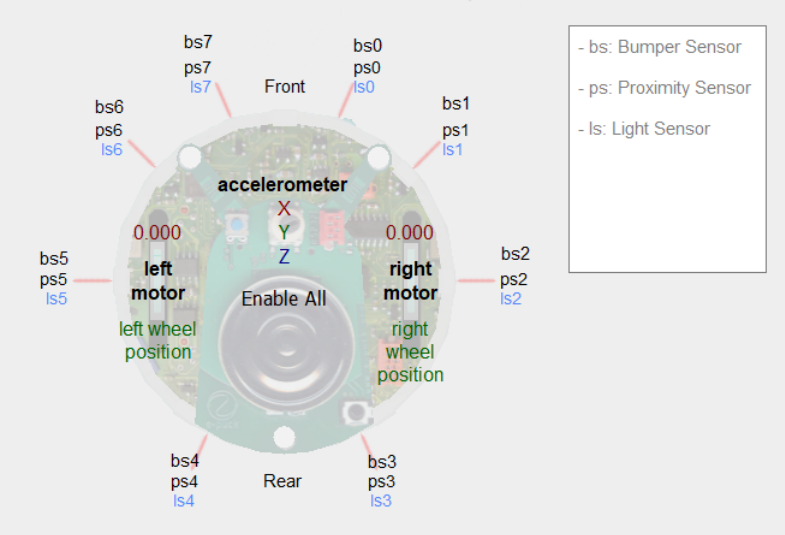


Figura 2: In [1] sono stati sviluppati 3 modelli di controller diversi che sfruttano diversamente (totalmente o parzialmente) l'embodiment presentato in figura. Nel caso di questo studio il modello di controller adoperato è il tipo `4`, nella quale i sensori posteriori di prossimità e collisione non vengono adoperati.

Lo studio sarà suddiviso nel seguente modo:

1. Nella parte iniziale verrà descritto in modo esaustivo quali sono le "azioni"[1] attese dal robot in relazione a possibili configurazioni di ostacoli e a quali pattern di attivazione ci si aspetta siano associate.

2. Successivamente si confronterà la situazione attesa sulla base delle nostre ipotesi con quella reale di un modello (già addestrato) tra quelli presentati in [1]. Il confronto verrà eseguito sulla base della traiettoria dell'agente nell'arena e dei pattern di attivazione presentati. Si focalizzerà l'attenzione solo su alcune sequenze di "azioni", anomale e non banali, impiegate dal robot per evitare gli ostacoli, confrontandole con le "azioni" attese sia per traiettoria che per pattern di attivazione neurale.

3. Si vuole, infine, comprendere come le varie attivazioni discrepanti ed anomale sono venute in essere, osservando l'evoluzione della rete neurale nel corso della fase di apprendimento.

[1] Sebbene non vi sia la definizione vera e propria del concetto di azione all'interno del modello.

2 Aspettative

In questa sezione verranno presentati i comportamenti attesi dall'agente in relazione ad alcune possibili configurazioni di ostacoli. Associate ai comportamenti saranno presentate anche le sequenze di attivazione che ci si attende di osservare una volta che l'agente abbia appreso il comportamento. Tali pattern sono espressi tramite un *"encefalogramma"* che descrive l'andamento delle attivazioni e disattivazioni dei neuroni, sfruttando dunque una *rappresentazione continua* piuttosto che esclusivamente *istantanea*.

Come anticipato nell'introduzione, le traiettorie che descritte dal comportamento e le rispettive sequenze di attivazione neurale possono essere previste sulla base del comportamento che l'agente assume se questi semplicemente collidesse con gli ostacoli dell'arena. In generale il comportamento dell'agente dinnanzi ad un ostacolo può essere pensato simile a quello di una palla da biliardo quando questa colpisce uno dei bordi del tavolo. *L'obiettivo dell'apprendimento è anticipare questi pattern di attivazione quando l'agente giunge in prossimità di un ostacolo* e, ovviamente, prima che questi vi collida. Il comportamento si suppone risulti simmetrico su entrambi i lati, ergo un ostacolo dovrebbe portare l'agente a presentare comportamenti simili e speculari sia che questo sia posto alla sua destra o alla sua sinistra.

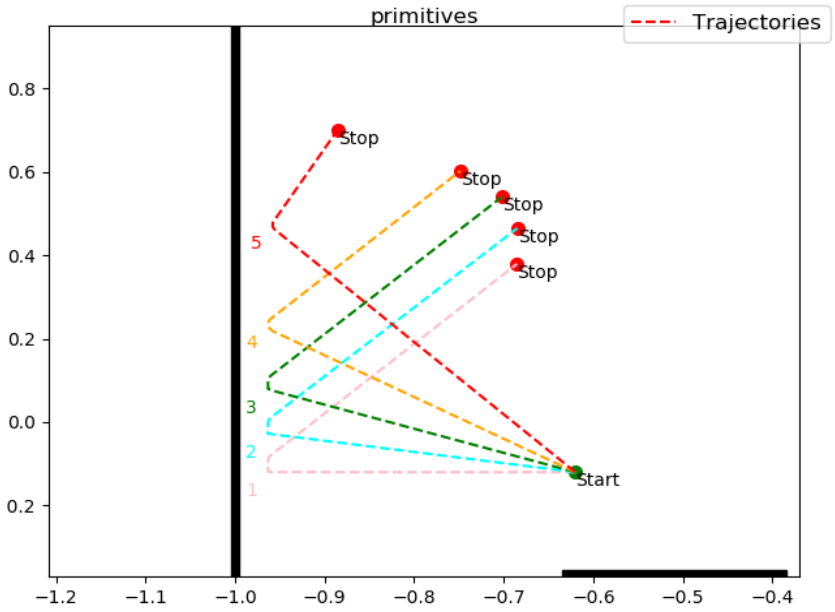


Figura 3: *Ostacolo Semplice*

La configurazione di ostacoli più semplice che l'agente può individuare è quella con un singolo ostacolo, in Figura 3 rappresentato dal muro dell'arena. In questo caso l'ostacolo è situato frontalmente e a sinistra dell'agente; nel caso l'ostacolo

fosse stato alla sua destra, il comportamento (individuato dalle traiettorie) sarebbe stato il medesimo ma speculare.

In Figura 3 sono individuate alcune possibili traiettorie che l'agente può seguire evitando l'ostacolo:

1. Se l'agente percorresse la traiettoria `1`, allora si ritroverebbe l'ostacolo posizionato perpendicolarmente al suo percorso. La *reazione attesa* in questo caso, solitamente, è quella di osservare il robot girare su stesso per poi tornare sui propri passi ma in questo caso l'agente sterza semplicemente a destra. Questo è dovuto in parte alla pura reattività del modello: se la potenza delle ruote non è sufficiente, l'agente non potrà completare una rotazione di 180°

2. Come si può invece vedere nelle traiettorie `2, 3, 4, 5`, maggiore è la porzione del corpo dell'agente esposta all'ostacolo[2] (e quindi maggiore il numero di neuroni attivi), maggiore sarà l'angolo di sterzata impiegato dall'agente .

Le traiettorie sopra esposte possono essere ricondotte a 3 principali pattern di attivazione neurale nel layer di collisione, visibili nelle Figure 4 e 5.

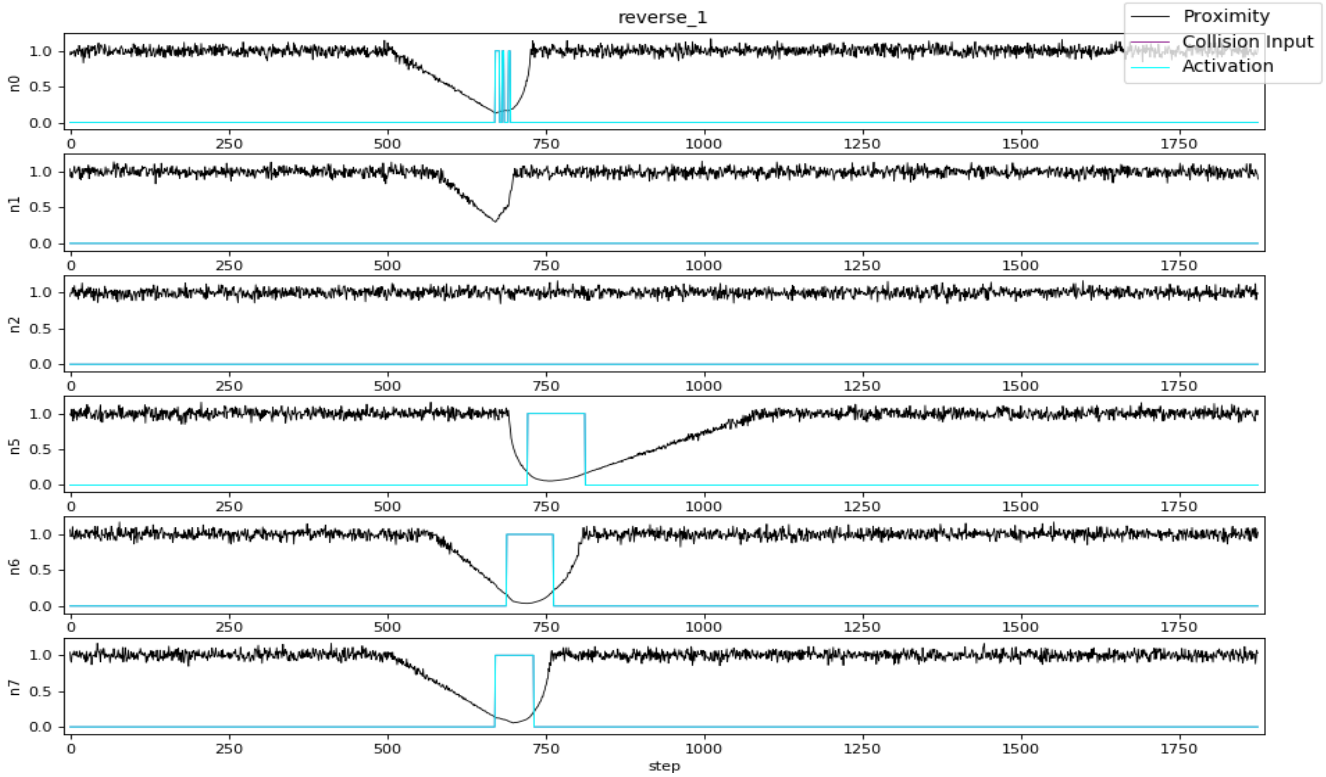
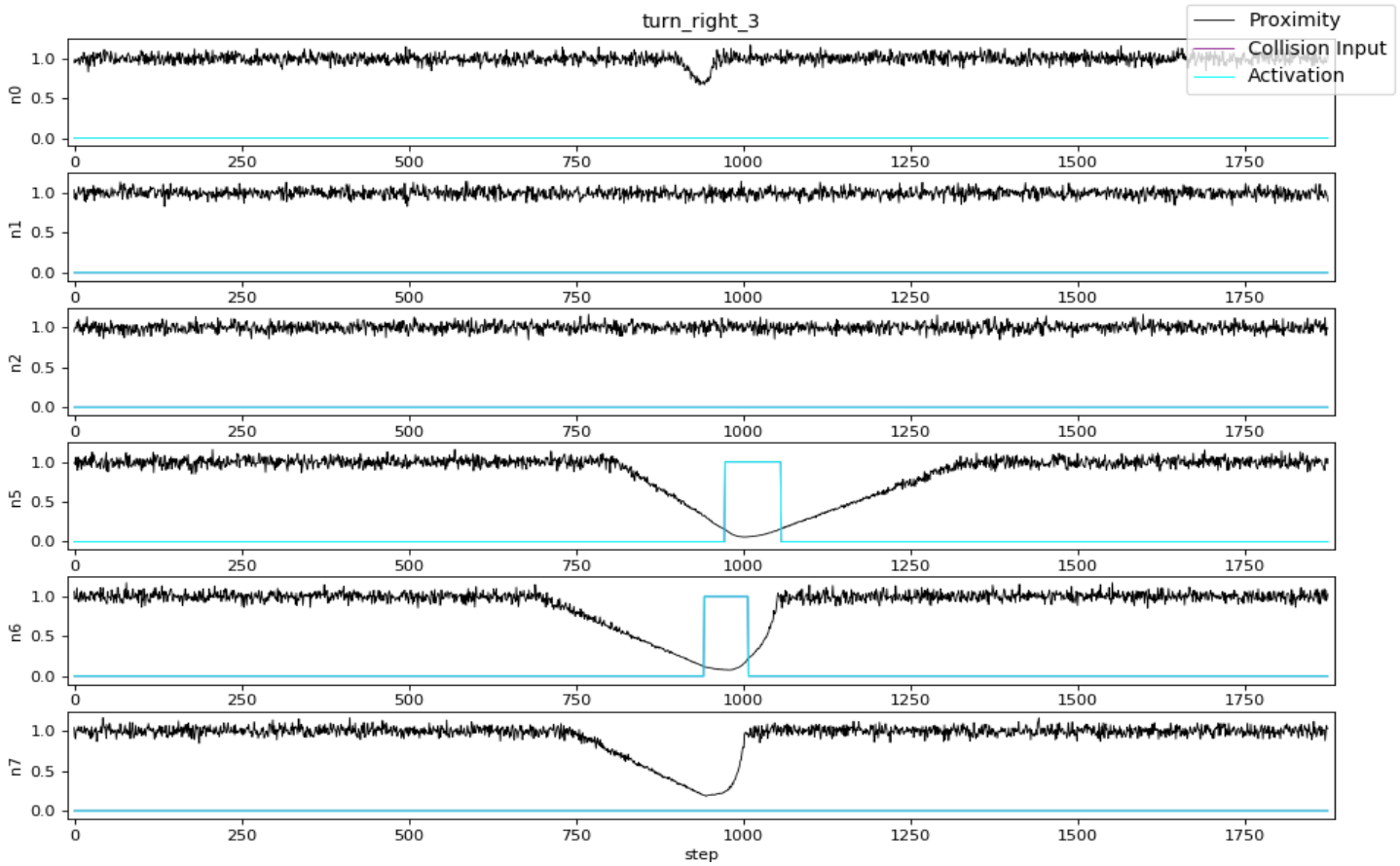
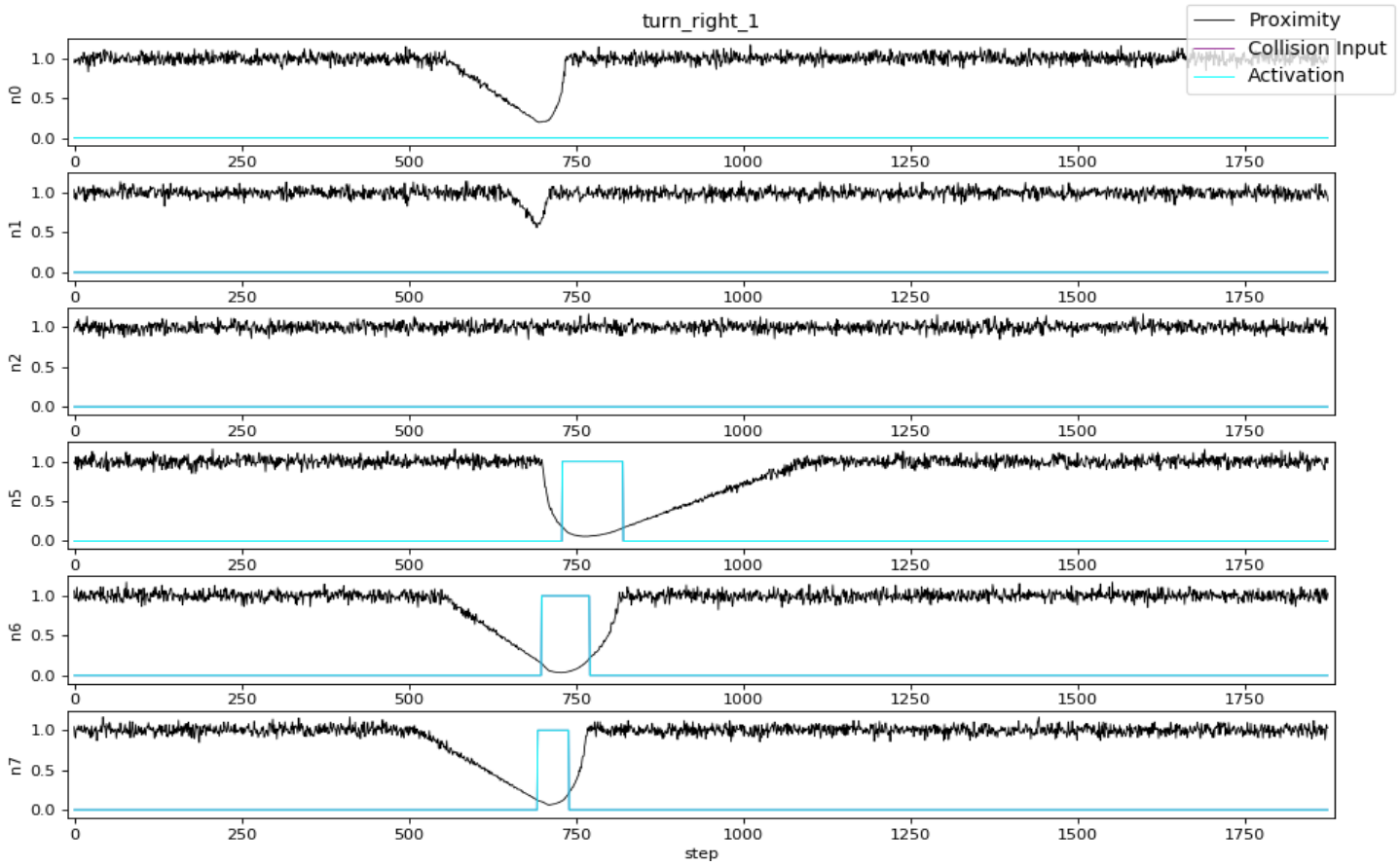


Figura 4: *Inversione del senso di rotazione della ruota destra*

[2]Ovvero il numero di sensori di prossimità vicini all'ostacolo



(a) *Accelerazione ruota sinistra – Mild*



(b) *Accelerazione ruota sinistra – Strong*

Figura 5: *Sterzata*

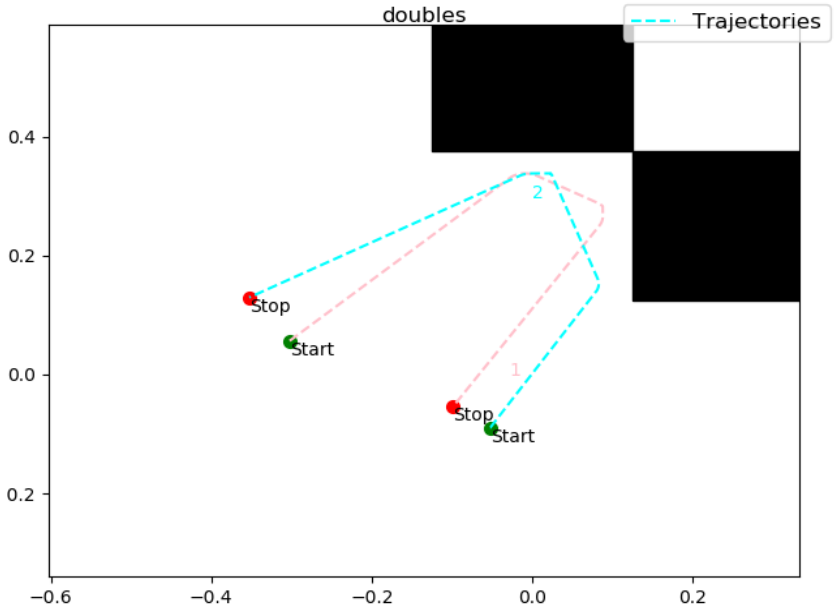
Il pattern di attivazione neurale visibile in Figura 4 è atteso qualora l'agente si trovi in una situazione simile a quella vista in Figura 3 – traiettoria `1`. La ruota scelta per l'inversione dipende da quali sono gli altri neuroni attivi nel layer di collisione. Se sono attivi più neuroni del lato destro rispetto a quello sinistro allora si inverte il senso nella ruota sinistra; viceversa, se sono attivi più neuroni nell'emisfero destro allora sarà la ruota destra a vedere invertito il proprio senso di rotazione. In caso di parità si inverte sempre il sensor della ruota destra.

Le sequenze di attivazione invece visibili nelle Figure 5a e 5b rappresentano le sequenze di attivazione attese dai neuroni di collisione qualora l'agente rilevi un ostacolo alla sua sinistra. Maggiore è il numero di neuroni attivi (contemporaneamente) in un certo lato, maggiore sarà la potenza erogata al motore per sterzare. Il numero di neuroni attivi dipende per la maggior parte dall'inclinazione con cui l'agente incrocia l'ostacolo, in quanto si espone un differente numero di sensori all'ostacolo. Potrebbero essere quindi attivi anche solo alcuni neuroni di un solo emisfero. Ad esempio, come si può vedere in Figura 5a, il pattern atteso nelle traiettorie `4` e `5` in Figura 3 possiede attivi solo i neuroni `5` e `6`, in quanto l'agente ha i soli sensori laterali rivolti verso gli ostacoli. La sequenza 5b rappresentata in figura è invece associabile alle traiettorie `2` e `3` visibili in Figura 3; in caso la svolta fosse stata a destra l'attivazione sarebbe stata simile ma sui neuroni dell'emisfero opposto (a parità di inclinazione della traiettoria).
In fase di apprendimento l'altro fattore che influisce sul numero di neuroni attivi durante un certa manovra sarà il peso delle connessioni neurali: se i pesi sono piccoli allora il livello di attivazione calcolato sarà insufficiente per superare la soglia di attivazione, lasciando il neurone inerte.

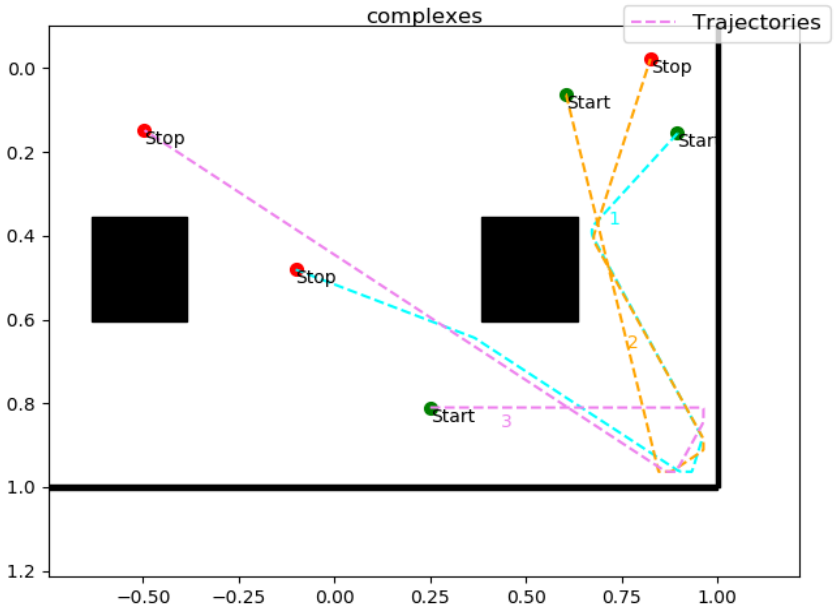
Come si è visto nei precedenti grafici, *la comparsa di queste sequenze non è istantanea bensì è un processo progressivo e graduale*: la sequenza d'attivazione partirà dai neuroni i cui sensori sono più vicini all'ostacolo, per poi progressivamente espandersi (front-to-back) verso gli altri neuroni man mano che viene eseguita la sterzata. Scomparirà infine allo stesso modo, partendo dai neuroni i cui sensori ora risultano i più lontani dall'ostacolo. Nel punto di massima vicinanza all'ostacolo ci si aspetta di osservare il maggior numero di neuroni attivi, in quanto l'agente deve sterzare violentemente al fine di non collidere con l'ostacolo.
Occorre, inoltre, fare una doverosa precisazione riguardo i pattern di attivazione presentati: nell'implementazione del controller dell'agente non è stato realizzato alcun meccanismo di mutua esclusione che coordini l'attivazione dei neuroni del Motor e Reverse Layer. Di conseguenza, quando l'agente sarà in fase di apprendimento, *non è da escludere che possano presentarsi pattern di attivazione che vedano attivi anche alcuni neuroni del lato opposto*, associabili alla presenza di

ostacoli ad ambedue i lati dell'agente. In generale si potrà presentare qualsiasi pattern risultante dalla composizione delle 3 sequenze di attivazione sopra illustrate.

Sulla base del comportamento dell'agente ipotizzato con la configurazione semplice degli ostacoli, è possibile farsi un'idea del comportamento atteso dall'agente con configurazioni di ostacoli più complesse, visibili in Figura 6.



(a) Ostacolo Doppio (Angolo Concavo)



(b) Impasse

Figura 6: *Composizione complessa di ostacoli*

Partendo dalla configurazione più semplice visibile in Figura 6a, questa presenta un angolo concavo formato dall'accostamento degli spigoli dei 2 ostacoli.
Nelle traiettorie `1` e `2` l'agente giunge rivolto in direzione di una delle 2 superfici che compongono la conca. Come si può vedere in entrambe le traiettorie, il comportamento atteso dall'agente è riconducibile alla composizione di più comportamenti "semplici"[3]. A livello di sequenze di attivazione, visibili in Figura 7, il comportamento può essere descritto nel seguente modo per ambedue le casistiche:

1. Si consideri la traiettoria `2`, Pattern 7b.

2. All'avvicinarsi dell'agente all'ostacolo, si attivano progressivamente i neuroni del lato destro. L'agente inizia la manovra di sterzata lentamente, a partire dal neurone mediano

   $$\{\texttt{0:Off, 1:On, 2:Off, 5:Off, 6:Off, 7:Off}\},$$

   accelerando progressivamente man mano che gli altri neuroni del lato si attivano

   $$\{\texttt{0:Off, 1:On, 2:On, 5:Off, 6:Off, 7:Off}\}.$$

   In un modello in fase di apprendimento nulla impedisce che si possa attivare anche il neurone frontale `0`, come si può osservare dalla ripida discesa dei valori dei sensori di prossimità in Figura 7b.

3. Ora l'agente si suppone stia percorrendo la "seconda componente" della traiettoria `2`. Alla collisione con il secondo ostacolo, vi è una progressiva attivazione del neurone frontale `0` e di tutti i neuroni laterali :

   $$\{\texttt{0:On, 1:On, 2:On, 5:Off, 6:Off, 7:Off}\}.$$

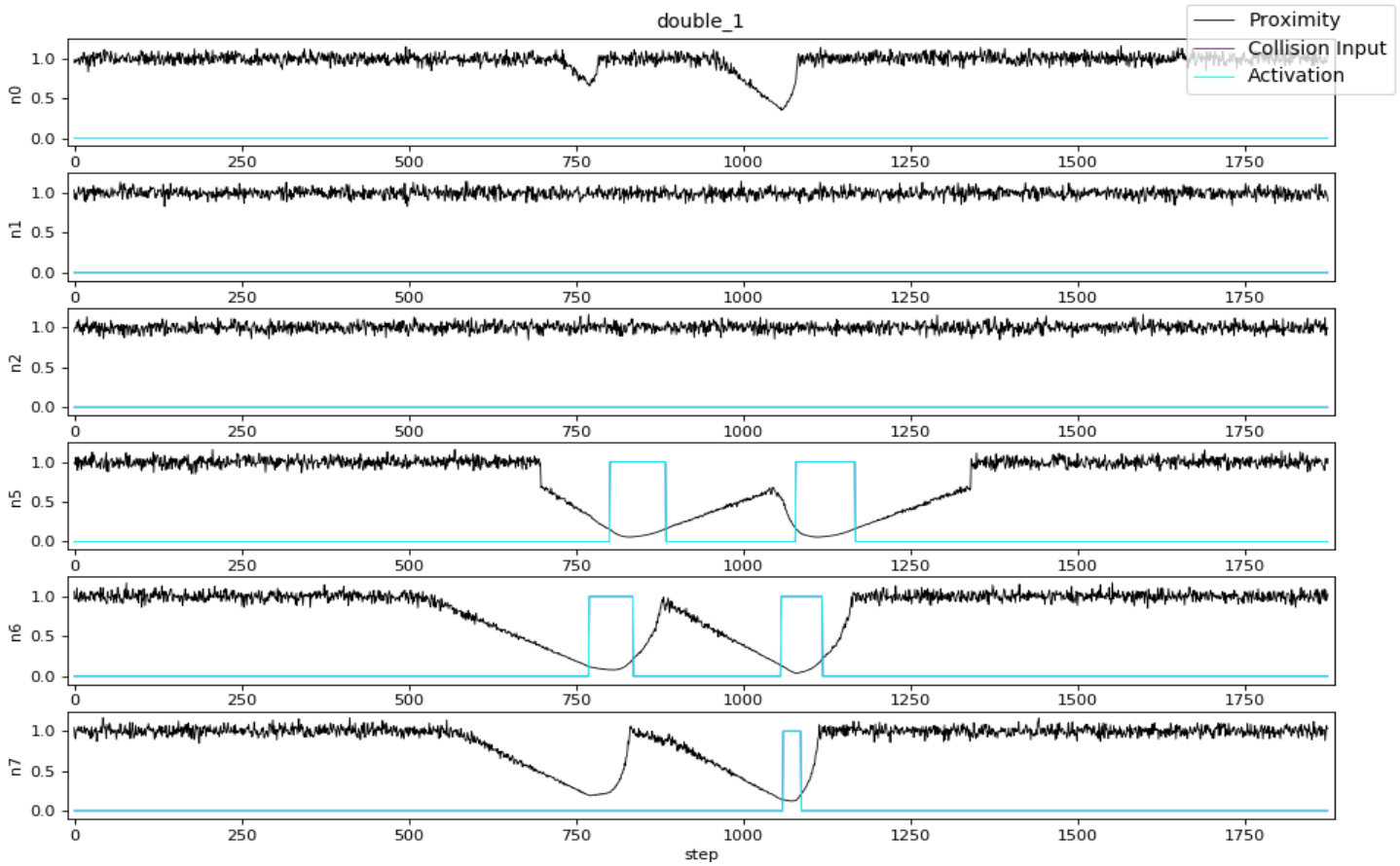
   Anche in questo caso, in un modello in fase di apprendimento, si sarebbe potuto osservare anche l'attivazione del neurone di collisione frontale `7`, data la vicinanza all'ostacolo. In tal caso ci si aspetta un brusca virata verso sinistra in quanto la ruota sinistra ha velocità non nulla e rotazione negativa!

4. Infine l'agente si troverà orientato verso la direzione indicata dalla terza componente della traiettoria `2`. In questa fase, durante l'addestramento della rete, potrebbero presentarsi delle attivazioni del neurone laterale di destra

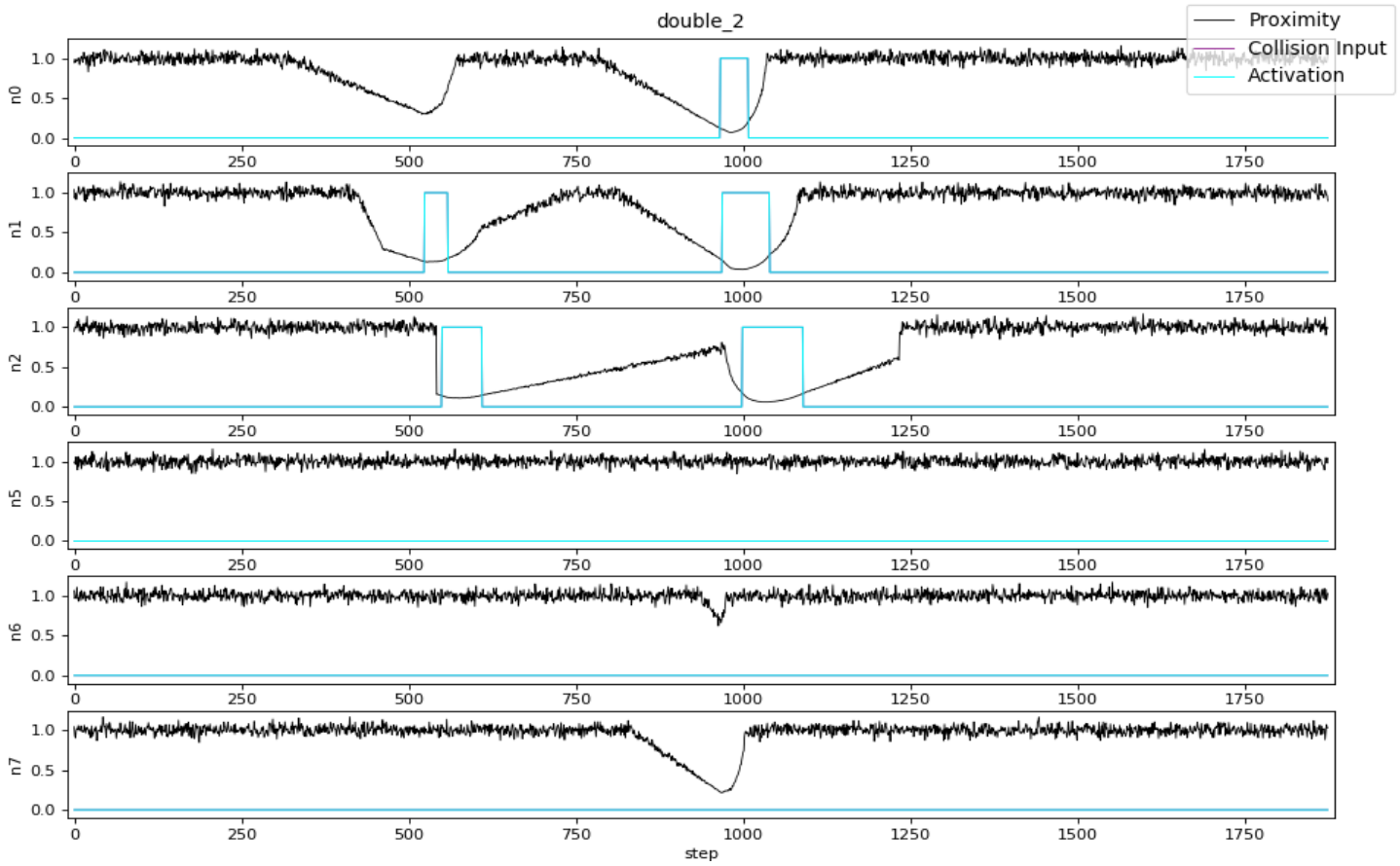
   $$\{\texttt{0:Off, 1:Off, 2:On, 5:Off, 6:Off, 7:Off}\}$$

   nelle aree di maggior vicinanza all'ultimo ostacolo, portando l'agente a stringere la traiettoria verso l'interno.

[3]Intesi come i comportamenti visti in Figura 3 ovvero su una configurazione semplice degli ostacoli.



(a) *Ostacolo Doppio (Angolo Concavo) – Sinistra*



(b) *Ostacolo Doppio (Angolo Concavo) – Destra*

Figura 7: *Sequenze di attivazione con doppio ostacolo*

In Figura 6b è invece visibile la seconda configurazione complessa di ostacoli. La posizione del blocco in relazione ai muri che delimitano l'arena crea 2 impasse, connessi ad angolo retto, nella quale l'agente potrebbe facilmente restare bloccato. Anche in questi casi il comportamento totale può essere visto come composizione di più comportamenti "semplici".

- Nella traiettoria `3` l'agente percorre una delle 2 strettoie in direzione parallela alle superfici degli ostacoli. Quando questi giunge al punto di giunzione dei 2 passaggi percepisce, perpendicolarmente alla sua posizione, il terzo ostacolo. La sequenza di attivazione sottostante, visibile in Figura 8, si presenterà inizialmente simile a quella mostrata in Figura 4. Dal momento però che l'agente è posizionato in corrispondenza dell'angolo della curva, quando questi ruota verso sinistra e procede con la manovra di sterzata, incontrerà il secondo muro dell'arena. La vicinanza dei due ostacoli provoca in rapida sequenza due pattern di attivazione che vede coinvolti tutti i neuroni del lato sinistro.

  In questo caso, dal momento che il sistema reagisce solo alla collisione con un ostacolo, non ci si aspetta che l'agente effettui una virata di 90 gradi al fine di imboccare la seconda strettoia. In fase di apprendimento, però, data l'implementazione dell'inversione di marcia (si veda pagina 7), l'agente potrebbe (potenzialmente) ruotare nella direzione con meno ostacoli e dunque a sinistra.

- Più complesse risultano invece le traiettorie `1` e `2`. Nella prima l'agente entra all'interno dell'impasse con direzione incidente all'ostacolo interno, riuscendo ad uscire dall'imboccatura opposta dopo aver superato il punto di incrocio dei muri dell'arena. Al contrario, nella seconda traiettoria l'agente giunge diretto verso l'ostacolo che chiude la curva, portandolo ad uscire dalla medesima strettoia da cui è entrato.

  In generale queste configurazioni sono una versione più complessa di quella vista in Figura 5a – traiettoria `2`. Si descrive di seguito il comportamento visibile in Figura 6b – traiettoria `2`, e il corrispondente pattern di attivazione 9b:

  1. Alla collisione dell'agente al muro dell'arena, si attivano progressivamente i neuroni del lato destro. L'agente inizia quindi lentamente la manovra di sterzata verso sinistra, a partire dal neurone frontale `0` per poi gradualmente attivare anche i neuroni laterali.

     `{0:On, 1:Off, 2:Off, 5:Off, 6:Off, 7:Off}`

     ↓

$$\{\texttt{0:On, 1:On, 2:Off, 5:Off, 6:Off, 7:Off}\}$$

$$\downarrow$$

$$\{\texttt{0:On, 1:On, 2:On, 5:Off, 6:Off, 7:Off}\}$$

2. Dal momento che l'agente si trova in prossimità del punto di giunzione dei muri dell'arena, si osservano in successione 2 rapide e consistenti attivazioni neurali sul medesimo emisfero. Questo porta l'agente in direzione della medesima apertura dalla quale è entrato.

3. Lungo la terza componente della traiettoria 2 l'agente è portato a sterzare leggermente a destra, attivando il neurone laterale 5

   $$\{\texttt{0:Off, 1:Off, 2:Off, 5:On, 6:Off, 7:Off}\},$$

   a causa della collisione con l'ostacolo interno.

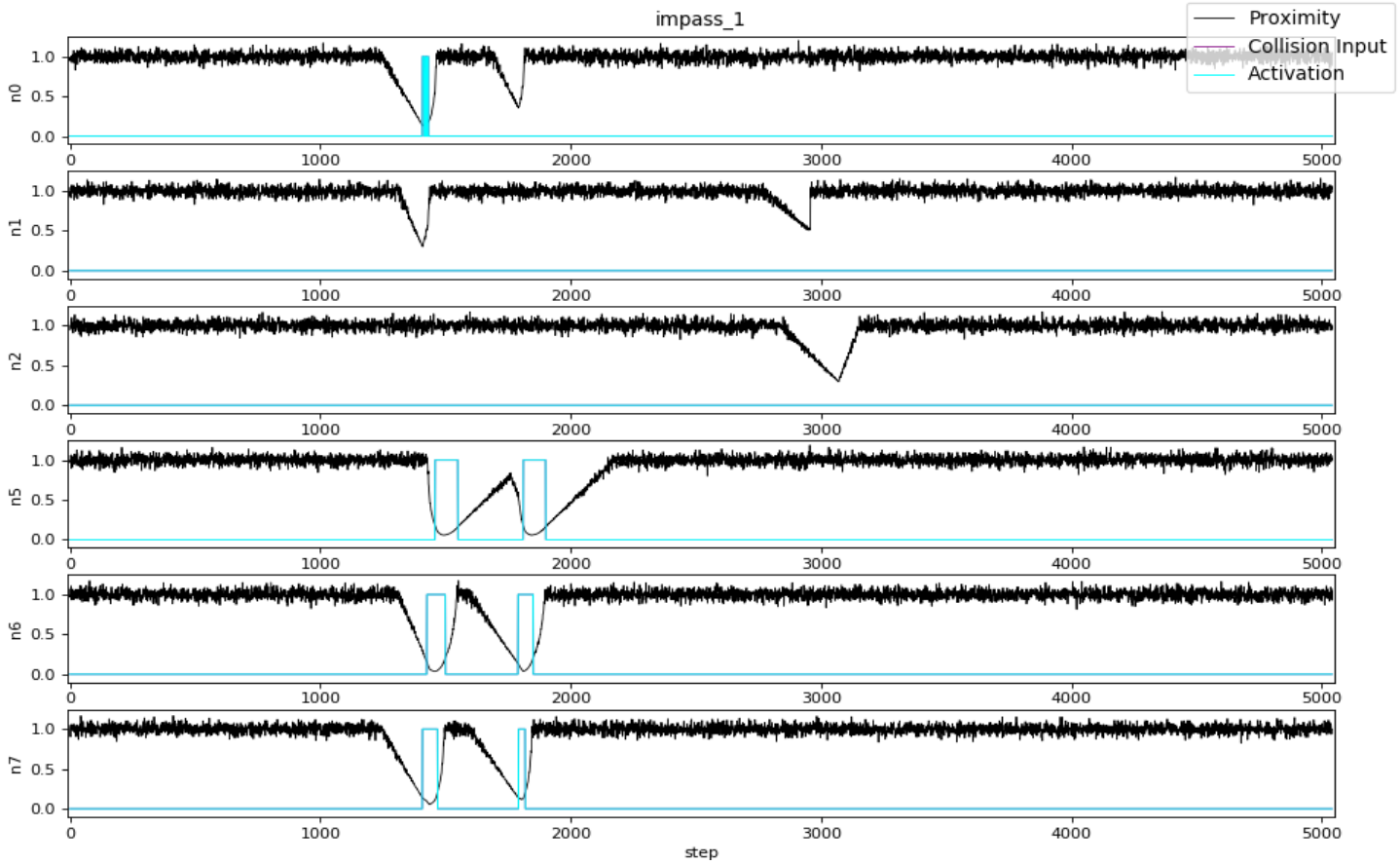
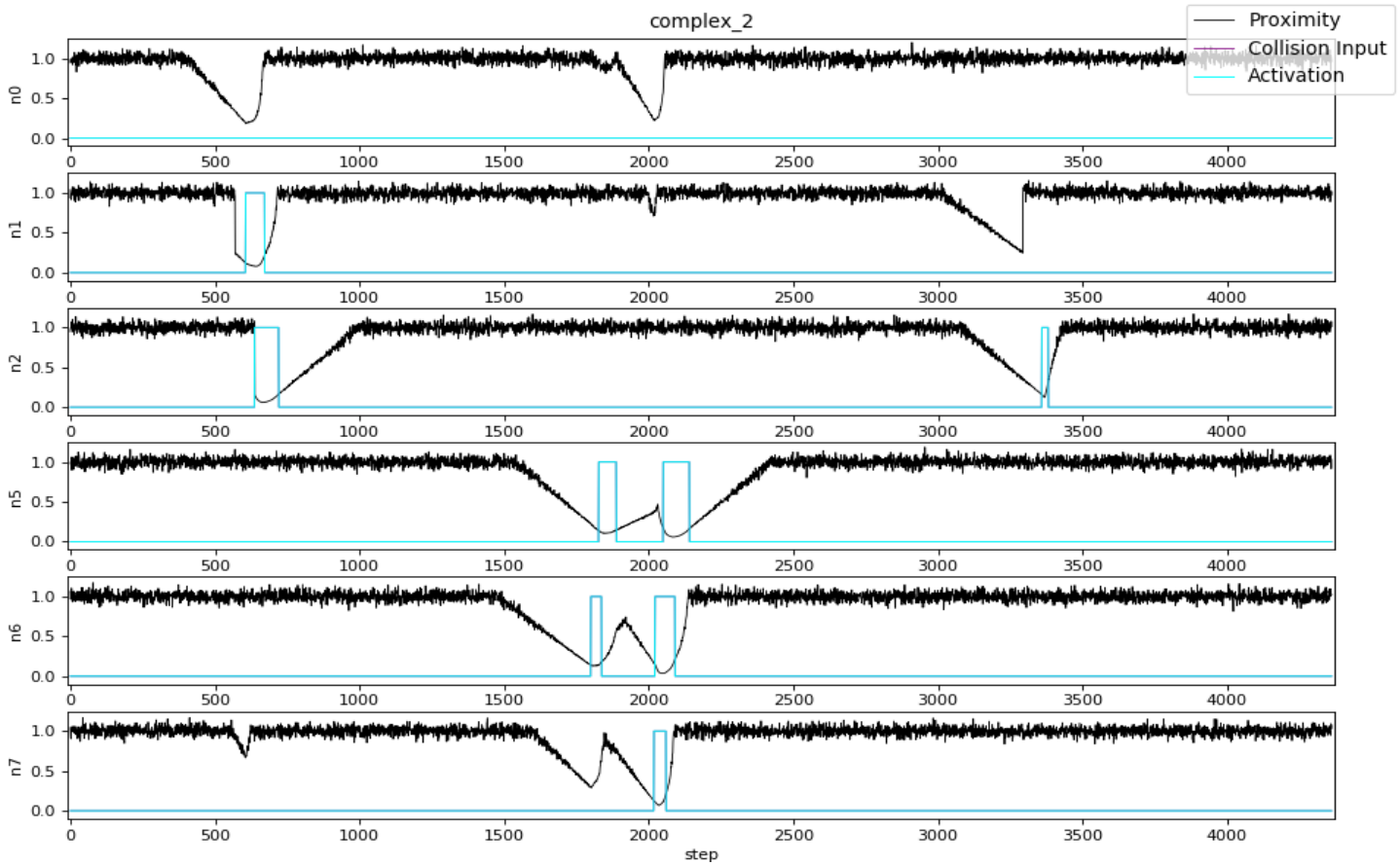
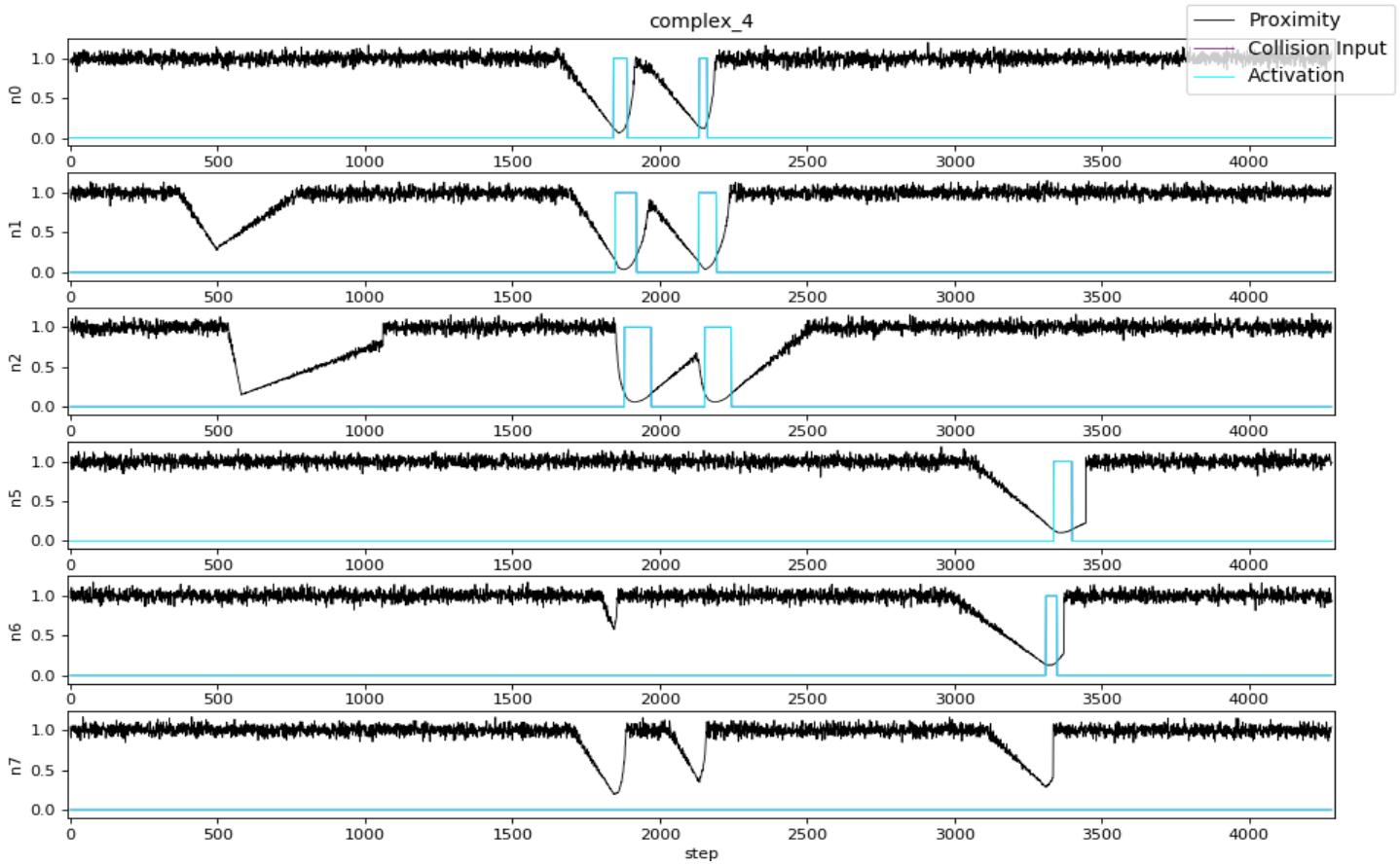


Figura 8: *Ostacolo Complesso – Impasse*



(a) *Ostacolo Complesso – 1*



(b) *Ostacolo Complesso – 2*

Figura 9: *Sequenze di attivazione con composizione complessa di ostacoli*

3 Risultati

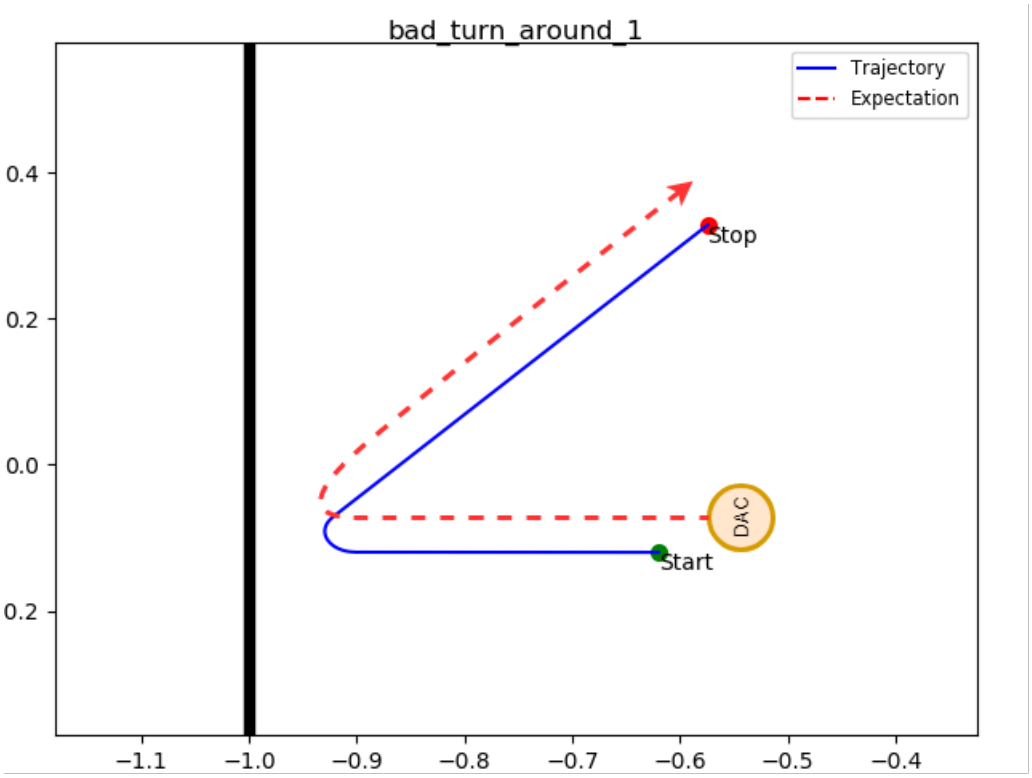
In questa sezione saranno presentati i comportamenti risultanti ottenuti attraverso un modello di agente (già) addestrato. L'agente è stato scelto in base alla classifica stilata nella sezione conclusiva di [1]. In questo caso è stato scelto il modello di tipo `4`, numero `361`, addestrato con i seguenti parametri di apprendimento $\{\eta$=`0.095`,ϵ=`0.7`,τ=`0.85`$\}$. La caratteristica del modello di tipo `4` è quella di non adoperare i sensori posteriori di prossimità e collisione.

Per ogni configurazione di ostacoli verranno presentate la traiettoria ottenuta, confrontata a quella prevista, ed un *"encefalogramma"*[4] del layer di Collisione della Rete Neurale, in relazione ai valori registrati dai sensori di Prossimità e i conseguenti valori in input agli stessi neuroni di Collisione. I valori di input sono il risultato della somma pesata di tutti i valori di output dei neuroni di Prossimità e i relativi pesi di connessione.

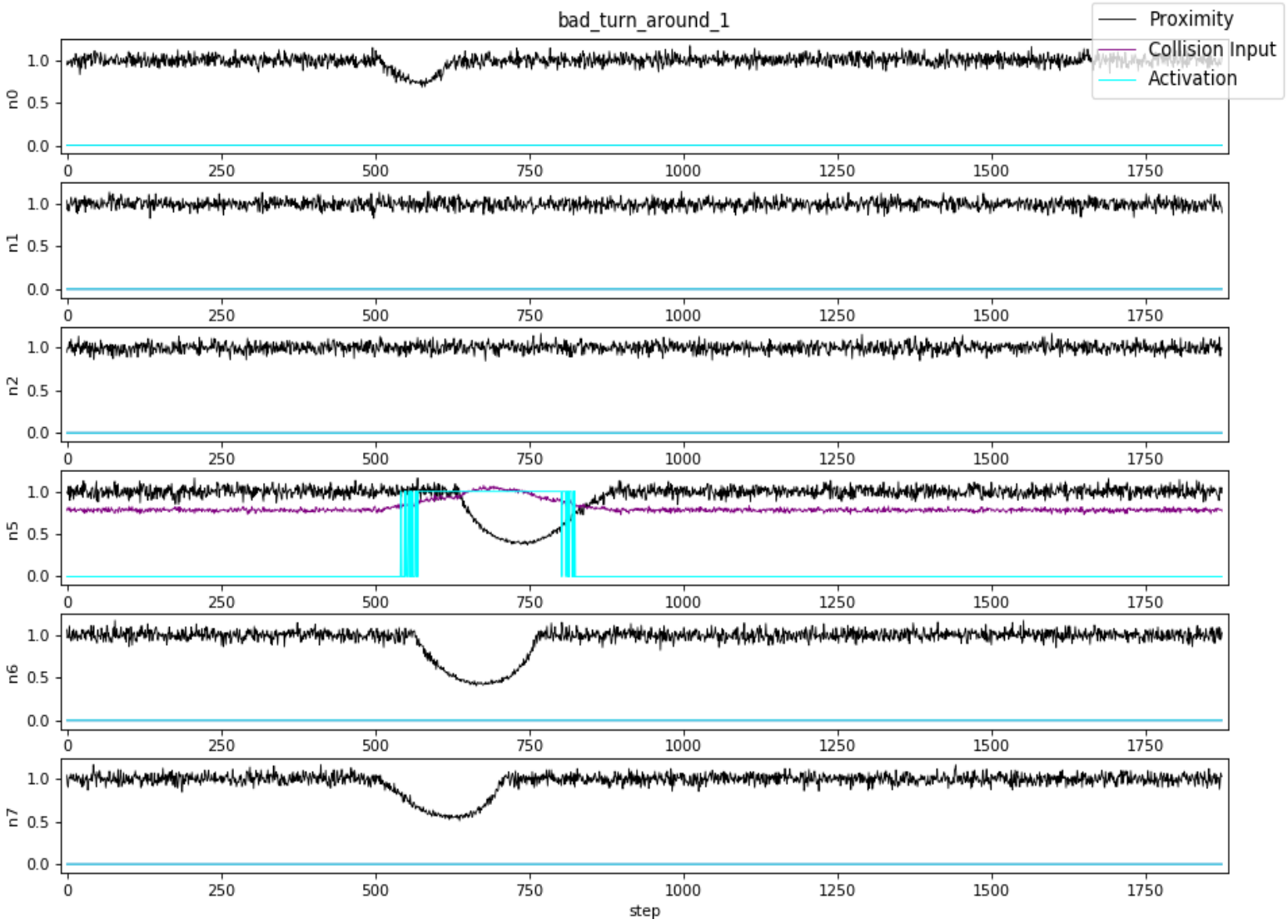
Iniziamo l'illustrazione a partire dai comportamenti ottenuti con le configurazioni semplici di ostacoli.

Come si può osservare nelle Figure 10a, 11a e 12a, dei 3 comportamenti "semplici" i soli a presentare la traiettoria attesa sono la svolta a destra e l'inversione di marcia [5]. Se però si osservano i relativi "encefalogramma" è possibile osservare come la sequenza di attivazione dei neuroni sia inaspettata: nella sezione precedente si era sottolineato come l'attivazione dei neuroni dovesse presentarsi in maniera graduale. Quello che si osserva invece in figura è che, dei 6 neuroni che compongono il layer di Collisione, se ne attiva sempre e solo uno, il numero `5`, che corrisponde al sensore di collisione laterale sinistro.
Sebbene nel caso della sterzata a destra (Figura 11) tale attivazione si sviluppa nel comportamento atteso, *il fatto che i neuroni più vicini all'ostacolo non si siano attivati è decisamente inaspettato ed anormale.* Inoltre, nel caso dell'inversione, l'agente non esegue affatto una manovra di inversione ma una semplice sterzata.
Anche nell'altro caso base, il fatto che sia sempre e solo lo stesso neurone ad attivarsi, porta l'agente in Figura 12a a "sbagliare" completamente la traiettoria di sterzata attesa, tornando invece indietro per la medesima direzione dalla quale è giunto anziché sterzare a sinistra.

[4] Andamento delle attivazioni $o_j = 1$ e disattivazioni $o_j = 0$ dei neuroni.
[5] come già fatto notare nella sezione Risultati di [1] ed anticipato nell'introduzione

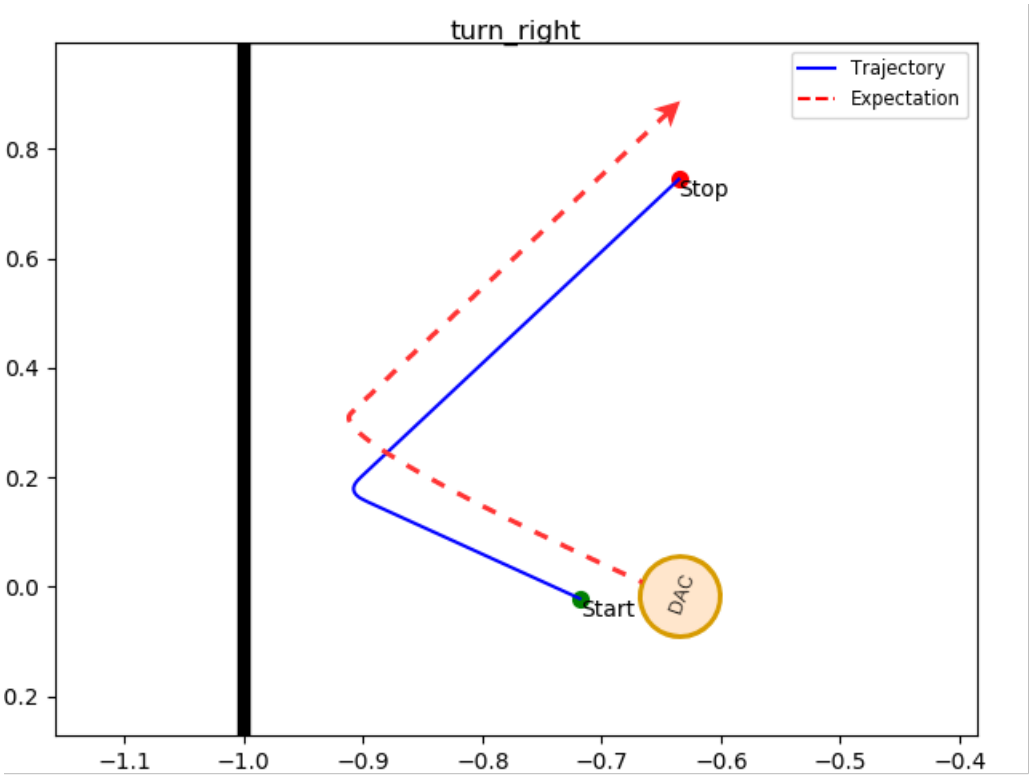


(a) Traiettoria

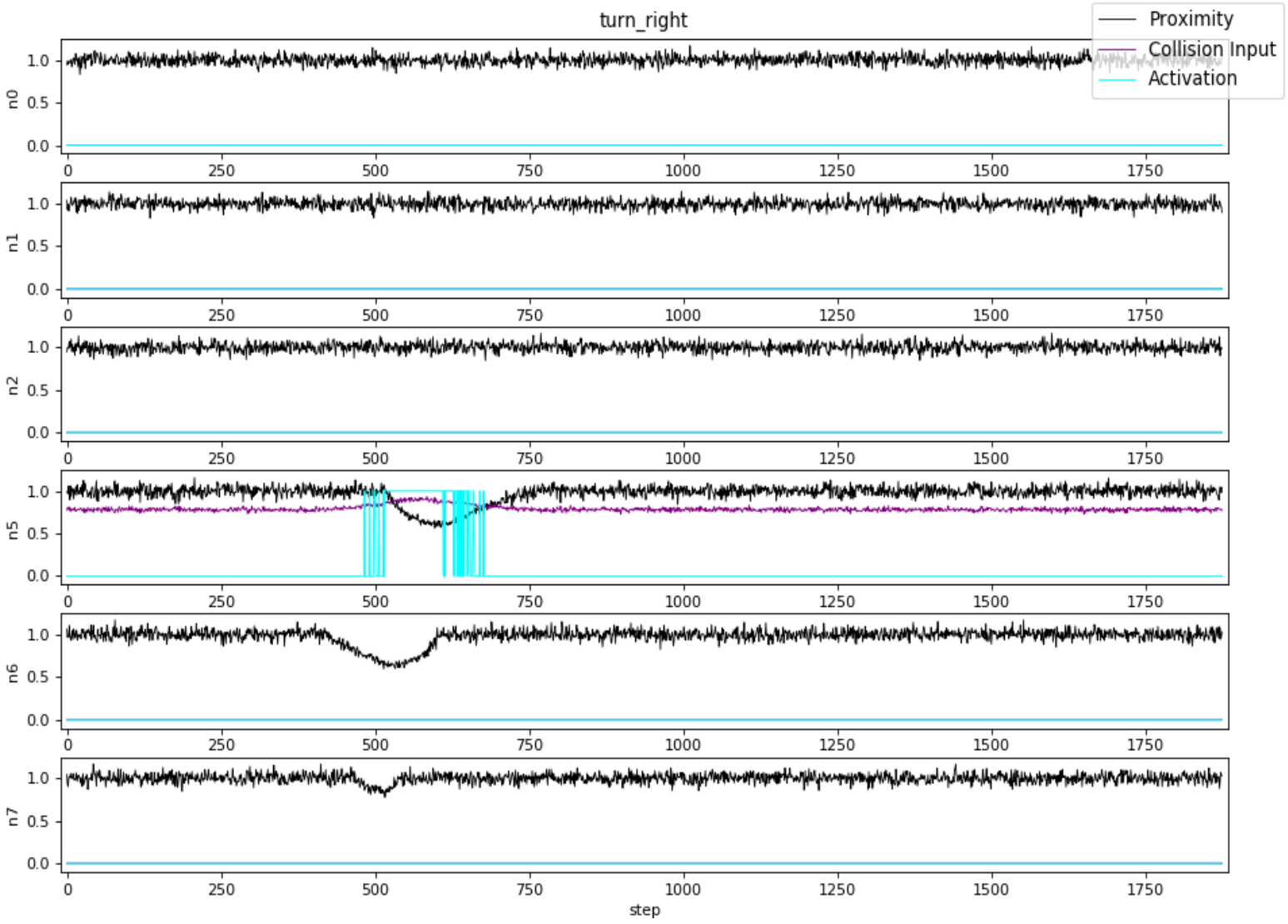


(b) "Encefalogramma"

Figura 10: Inversione di Marcia

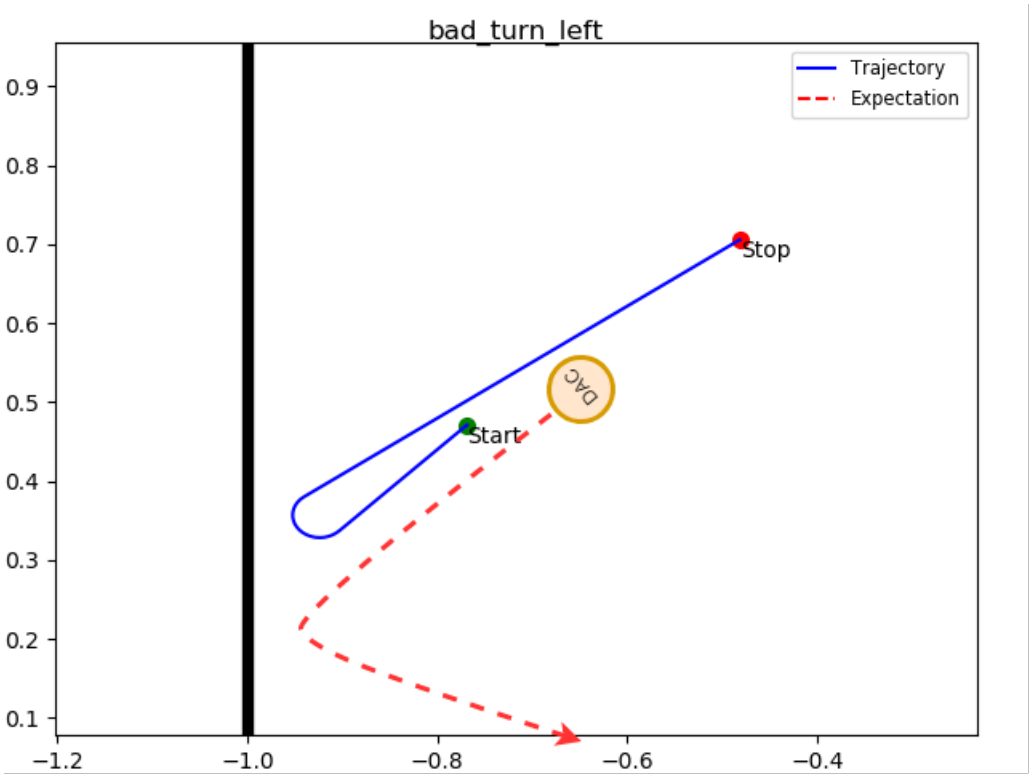


(a) Traiettoria

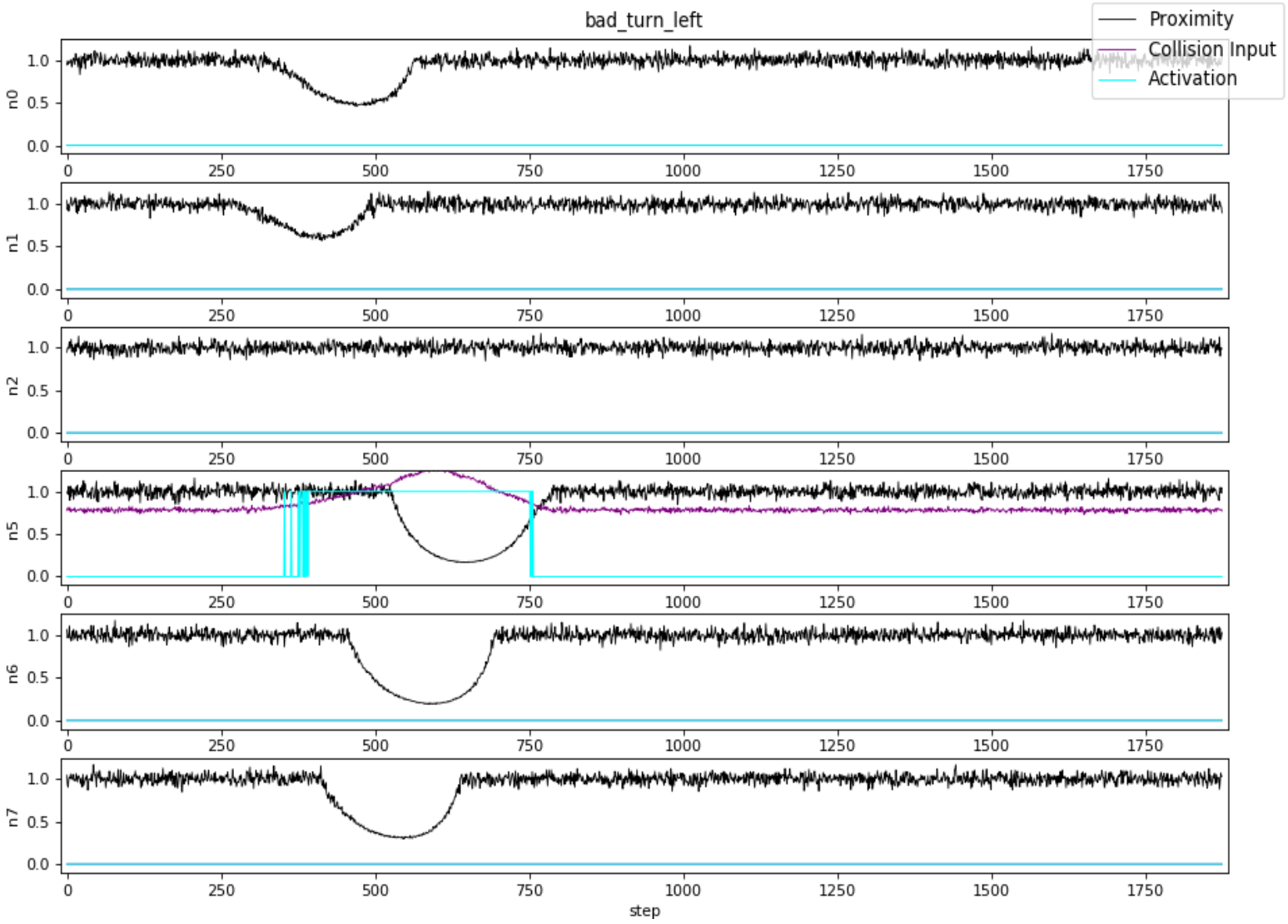


(b) "Encefalogramma"

Figura 11: Sterzata a destra



(a) Traiettoria



(b) "Encefalogramma"

Figura 12: Sterzata a Sinistra

Osservando attentamente i 3 "encefalogramma" 10b, 11b e 12b, è possibile notare come, in tutti, la fase di attivazione è anticipata rispetto al reale incremento di prossimità[6], rappresentato dall'avvallamento nei valori del sensore di prossimità.
Se si osserva in maniera ancora più attenta, si vede come tale attivazione avviene in corrispondenza degli avvallamenti dei sensori frontali di prossimità, sebbene i neuroni associati a questi non si siano attivati.
Sebbene fosse prevedibile un'influenza da parte degli altri valori percepiti dai sensori di prossimità per l'attivazione di un singolo neurone di collisione, in quanto insita nel modello stesso[7], l'eccessiva sensibilità dimostrata è da ricondurre ad uno sbilanciamento dei pesi che il processo di apprendimento ha assegnato alle connessioni neurali.

Riportiamo di seguito la matrice di connessione tra il layer di Prossimità ed il layer di Collisione:

	cn0	cn1	cn2	cn5	cn6	cn7
pn0	$6.776e-172$	$1.116e-172$	0.0	0.303	$5.79e-21$	$3.122e-24$
pn1	$3.794e-172$	$9.474e-173$	0.0	0.301	$5.768e-21$	$2.514e-24$
pn2	$3.060e-172$	$4.364e-173$	0.0	0.304	$5.904e-21$	$2.459e-24$
pn5	$3.430e-172$	$4.430e-173$	0.0	0.533	$1.343e-20$	$2.462e-24$
pn6	$6.815e-172$	$7.849e-173$	0.0	0.361	$1.351e-20$	$3.936e-24$
pn7	$7.602e-172$	$1.054e-172$	0.0	0.317	$1.005e-20$	$5.897e-24$

Tabella 1: La tabella riportata è da leggersi nel seguente modo: nelle singole celle sono riportati i pesi di una connessione neurale, la connessione neurale a cui fa riferimento il peso è indicata all'inizio di ogni riga (neurone di prossimità) e nel header della colonna (neurone di collisione). Perciò il neurone di prossimità `pn0` è connesso al neurone di collisione `cn5` con peso 0.303.

Come è possibile evincere dalla matrice delle connettività, le uniche connessioni che possiedono un peso di dimensione sensibile sono, appunto, quelle relative al neurone di collisione `5`. Tutte le altre connessioni, nel corso del processo di apprendimento, hanno acquisito dei valori talmente piccoli da poter essere considerate nulle; nel caso del neurone di collisione numero 2, il valori dei pesi è esattamente 0.0. Questi valori, moltiplicati per gli output dei neuroni di prossimità $o_n = e^{-ps_n}, o_n \in [0.3679, 1.0]$, portano a valori che, sebbene vengano sommati

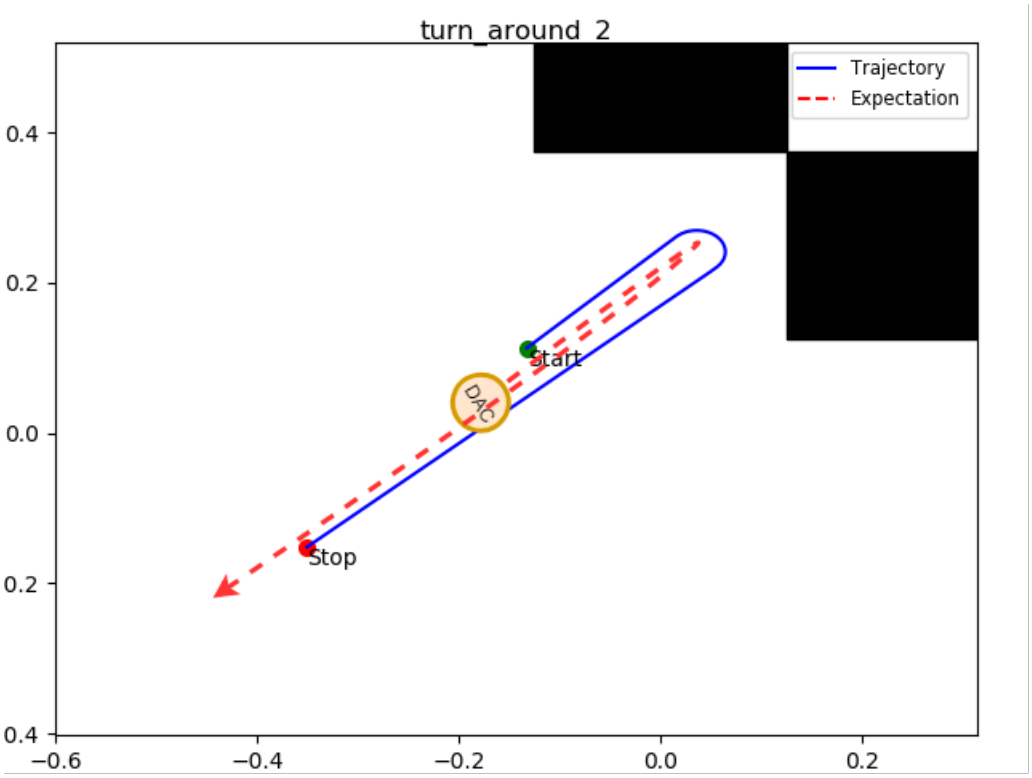
[6] I valori sono quelli percepiti dal sensore di prossimità `5`. Si veda Figura 2 come riferimento.
[7] Tutti i neuroni del layer di Prossimità sono connessi a tutti i neuroni del layer di Collisione.

per il calcolo del livello di attivazione dello stesso neurone, risultano sempre troppo piccoli, impedendo il superamento della soglia di attivazione $\tau = 0.85$ e di conseguenza l'attivazione del neurone. I valori delle connettività per il neurone 5 sono invece tali da permettere di attivarsi sia quando (apparentemente) necessario, sia quando non lo è. Infatti, dal momento che tutti i neuroni ricevono feedback calcolati dall'aggregazione delle percezioni di tutti i sensori di prossimità, se i pesi di un neurone sono abbastanza grandi, vi è la possibilità che questo si attivi sebbene l'ostacolo sia sul lato opposto.

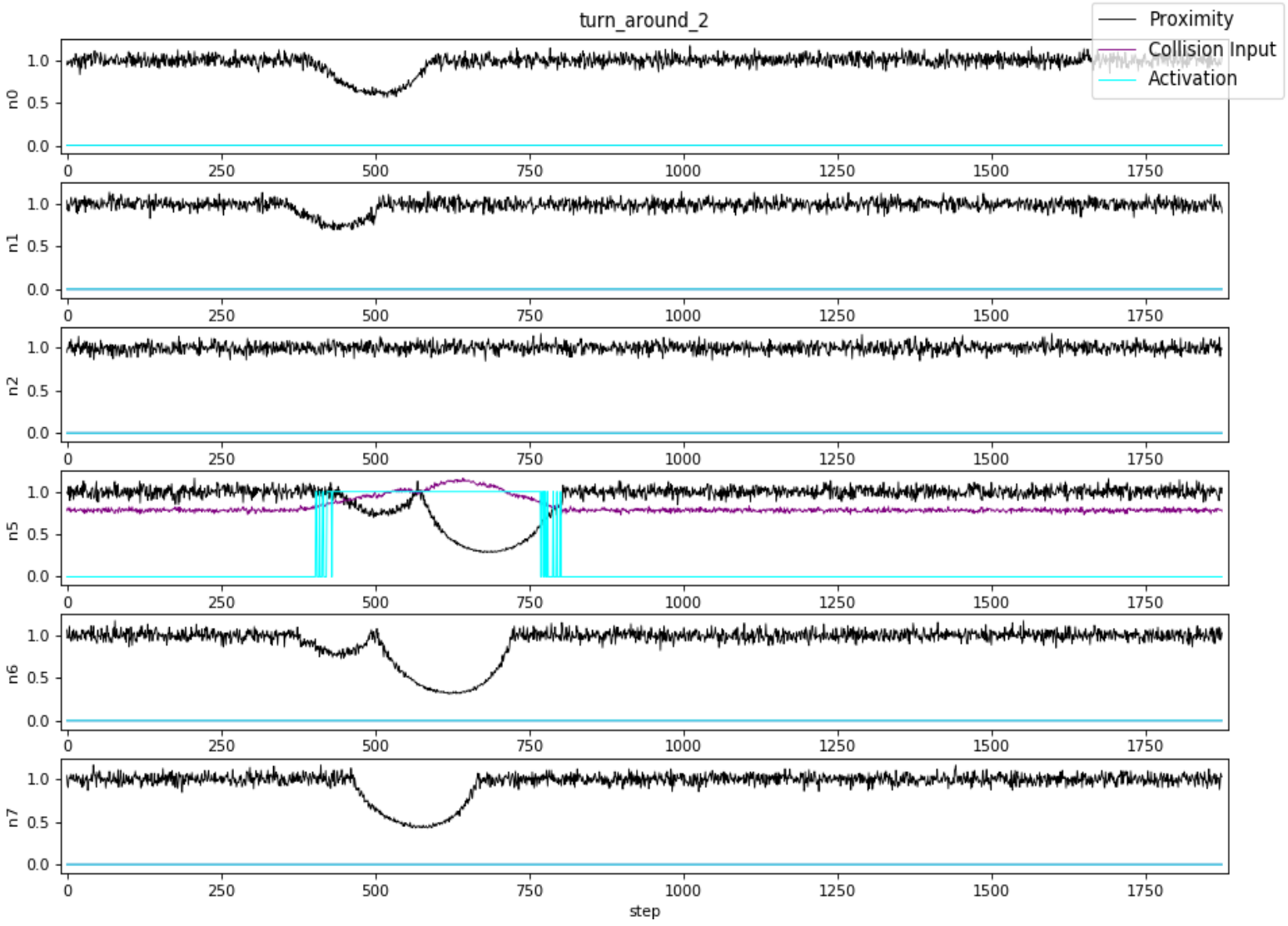
Si riportano nelle prossime pagine i risultati dei comportamenti ottenuti con le configurazioni di ostacoli complesse, accoppiate ai relativi "encefalogramma". I risultati ottenuti, alla luce di quelli osservati nei comportamenti "semplici", non soddisfano ovviamente le aspettative se non nei casi in cui l'unico emisfero coinvolto sia il sinistro. Ma anche in questi casi si presentano dei pattern di attivazione ben lontani da quelli prognosticati.

In Figura 13a è presentata una configurazione di ostacoli già studiata nella sezione precedente ma cui la traiettoria non era stata contemplata. Il motivo di tale mancanza è dovuta al fatto che l'agente, sfruttando le sole collisioni, semplicemente si blocca nel punto di giunzione dei 2 ostacoli. Sebbene il comportamento appreso non sia affatto perfetto, il fatto che l'agente sia stato comunque in grado di gestire una situazione non risolvibile usando il semplice movimento reattivo si può dire degna di nota.

In Figura 18b sono visibili delle attivazioni anche in altri neuroni oltre al numero 5. Questo è dovuto al fatto che l'agente toccando l'ostacolo attiva automaticamente il neurone relativo al sensore di collisione coinvolto.

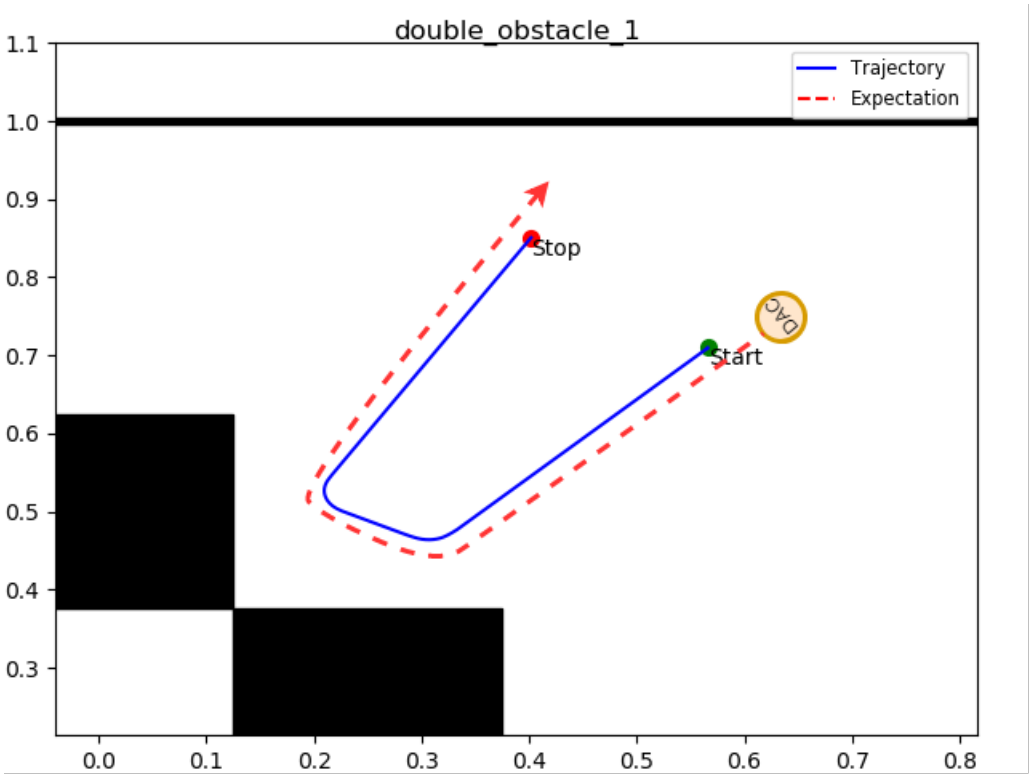


(a) Traiettoria

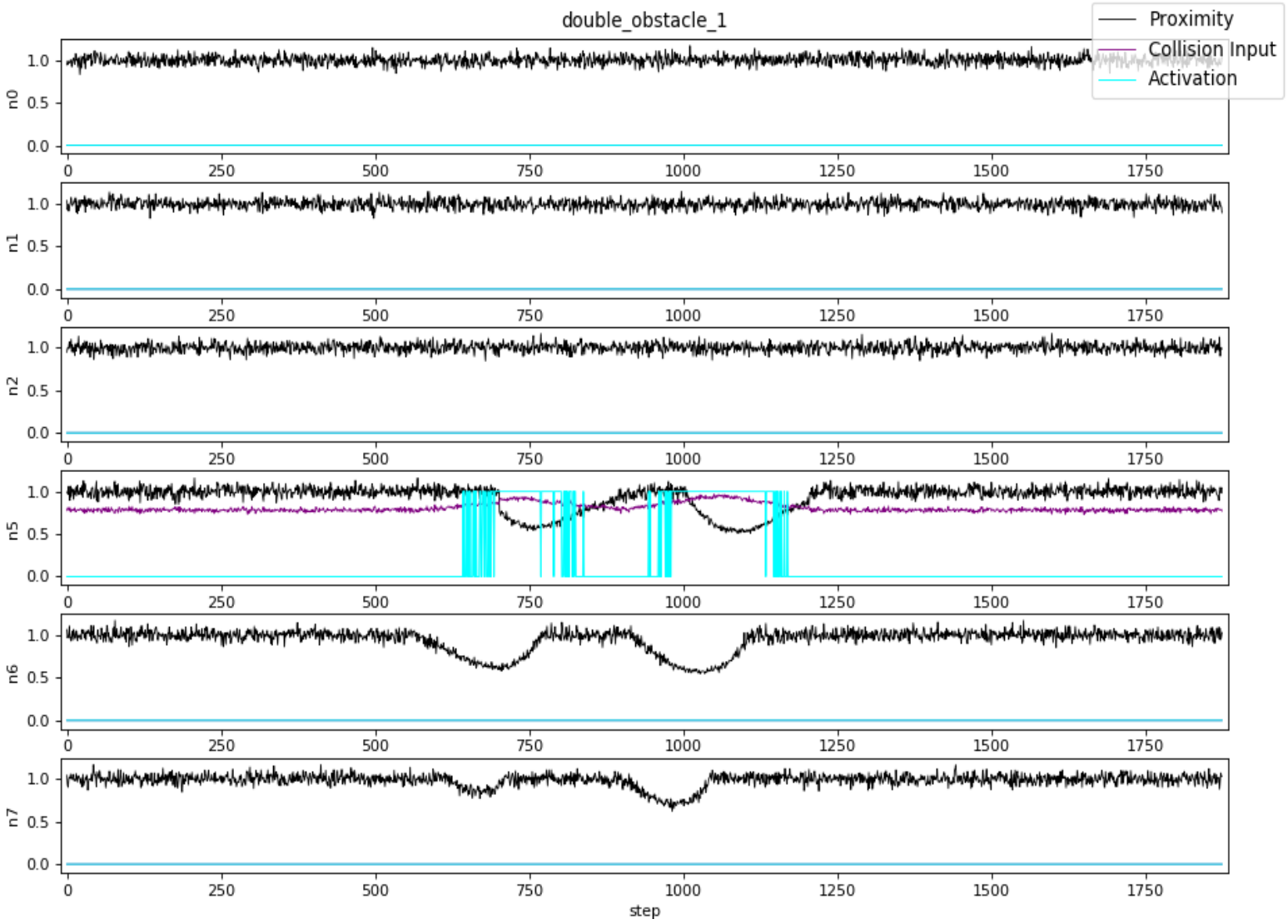


(b) "Encefalogramma"

Figura 13: Inversione di Marcia con doppio ostacolo

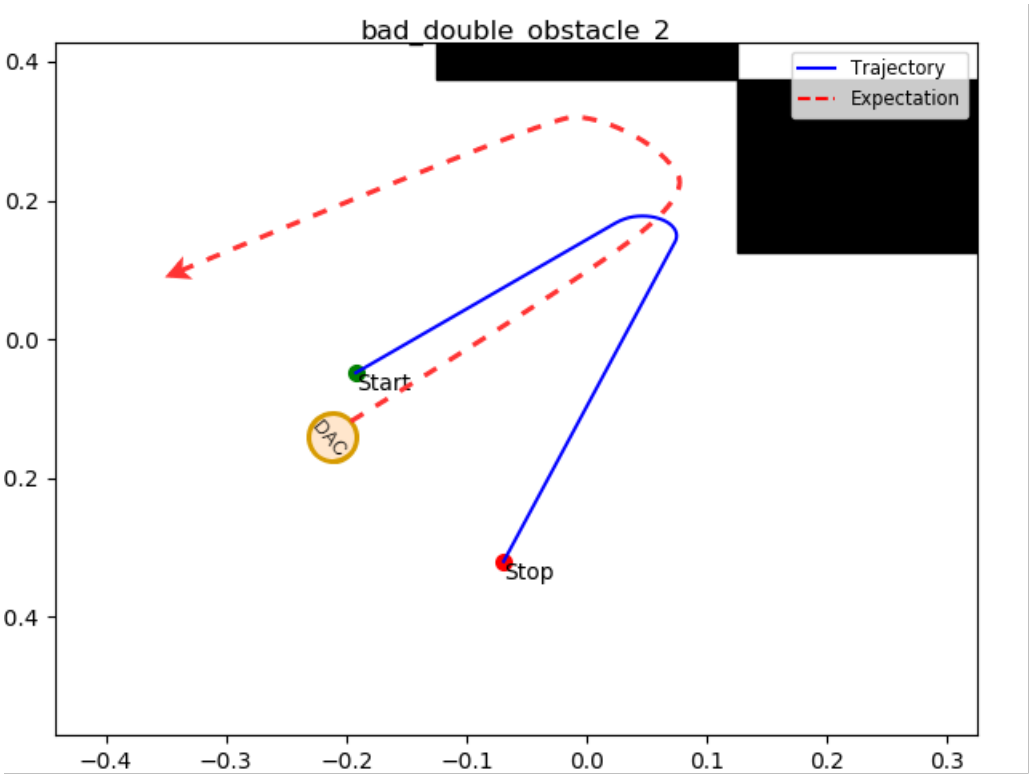


(a) Traiettoria

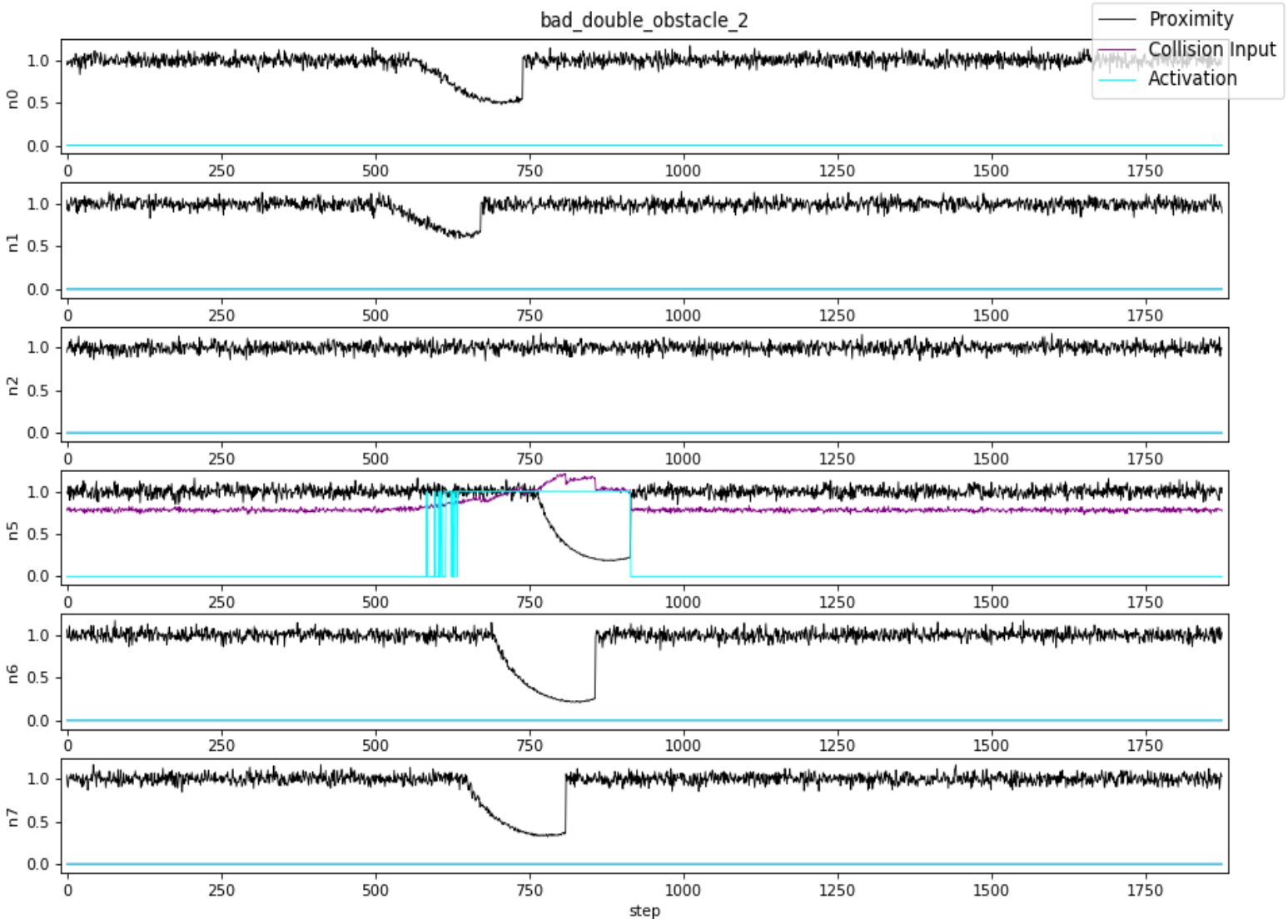


(b) "Encefalogramma"

Figura 14: Doppio ostacolo (sinistra)

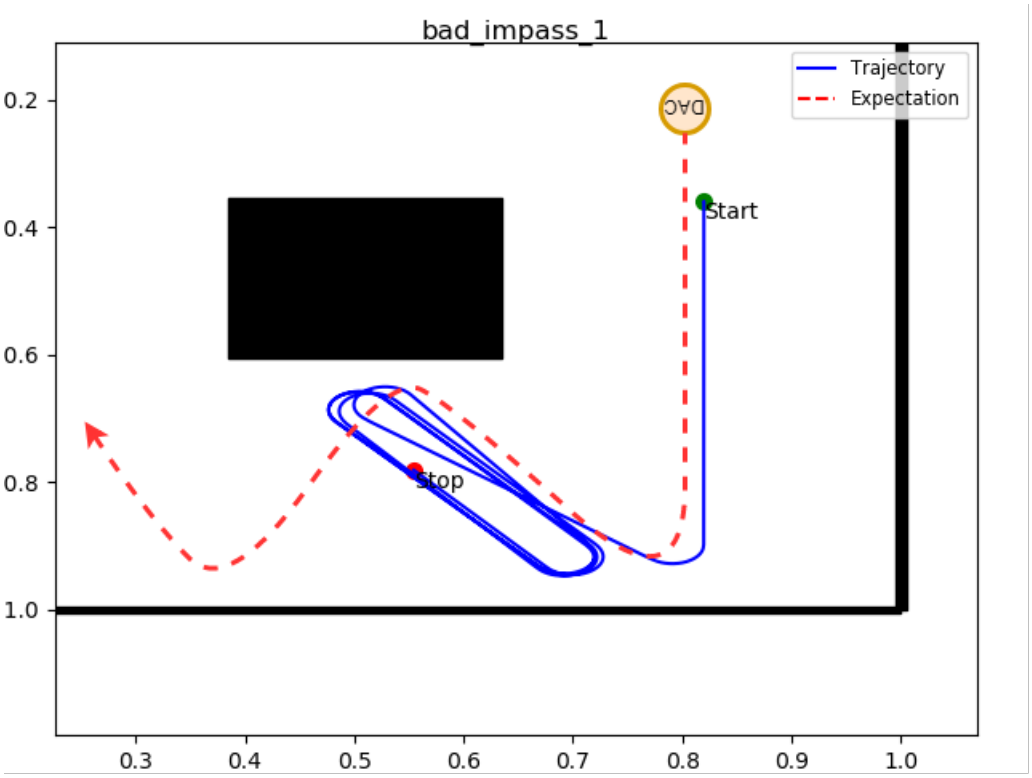


(a) Traiettoria

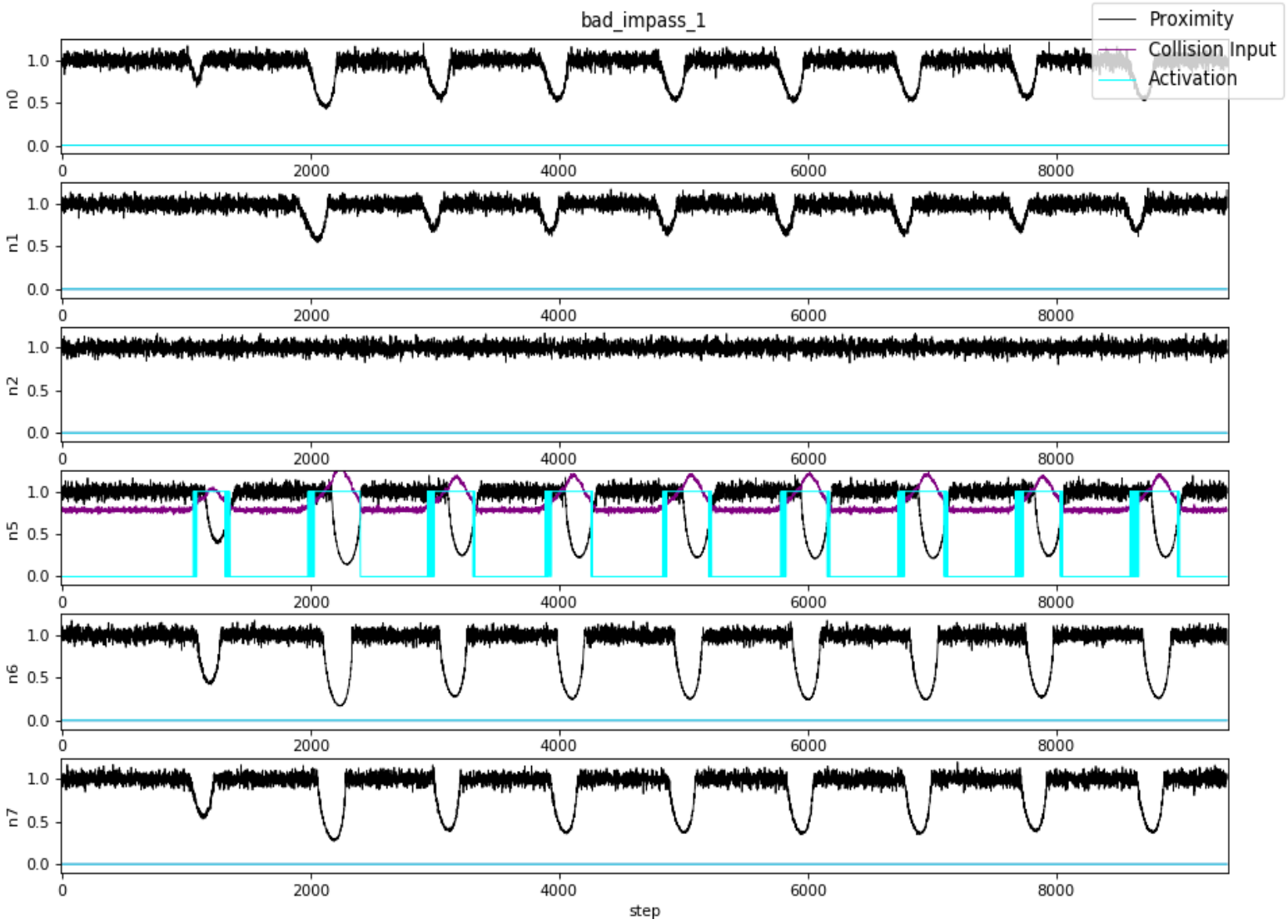


(b) "Encefalogramma"

Figura 15: Doppio ostacolo (destra)

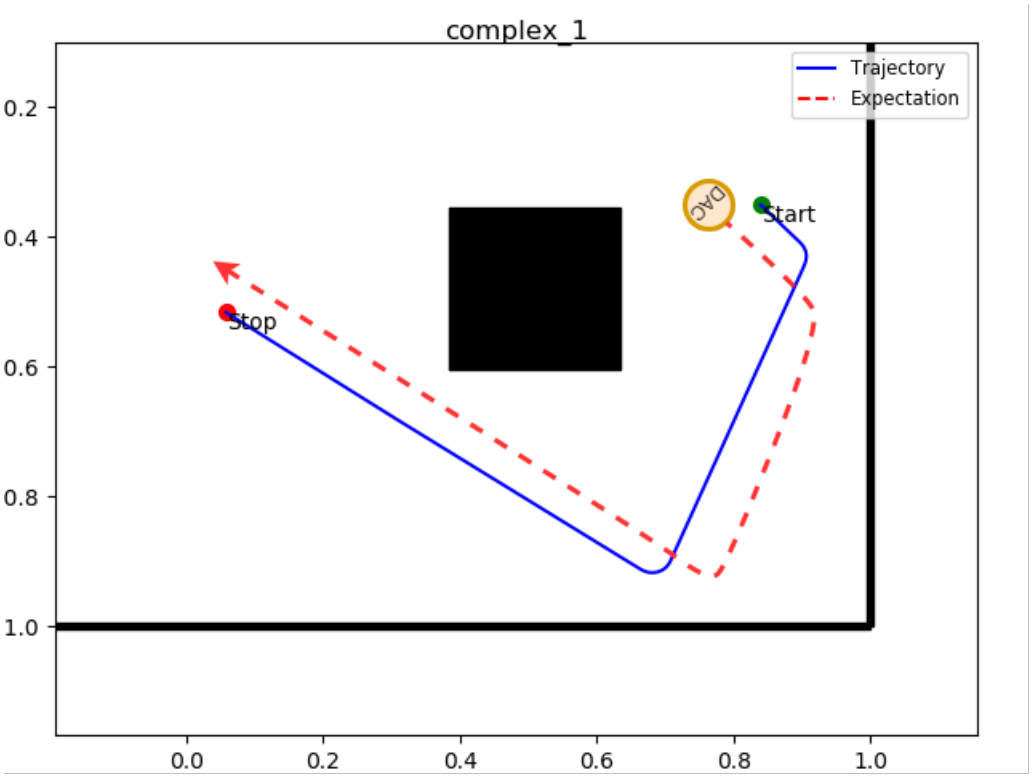


(a) Traiettoria

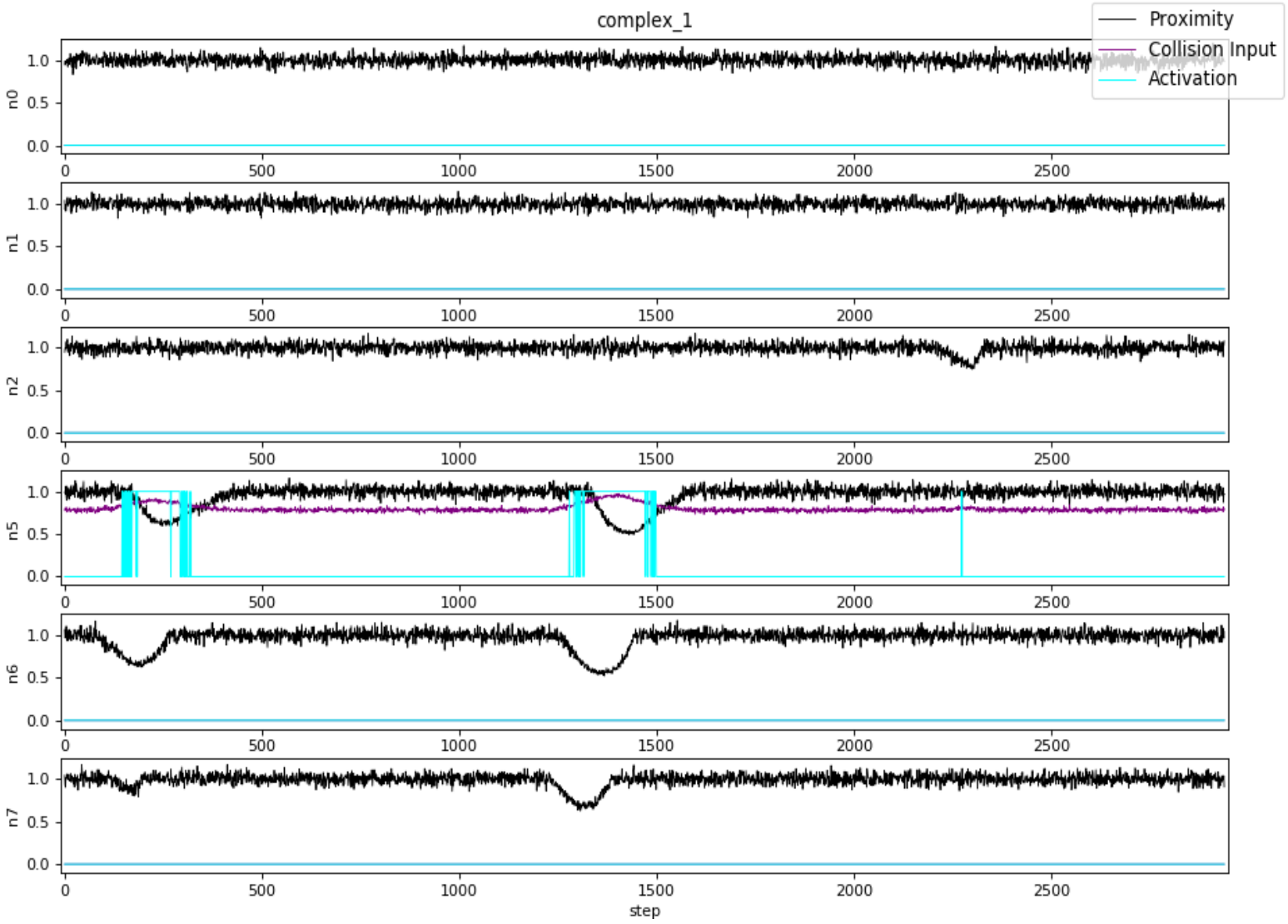


(b) "Encefalogramma"

Figura 16: Impasse

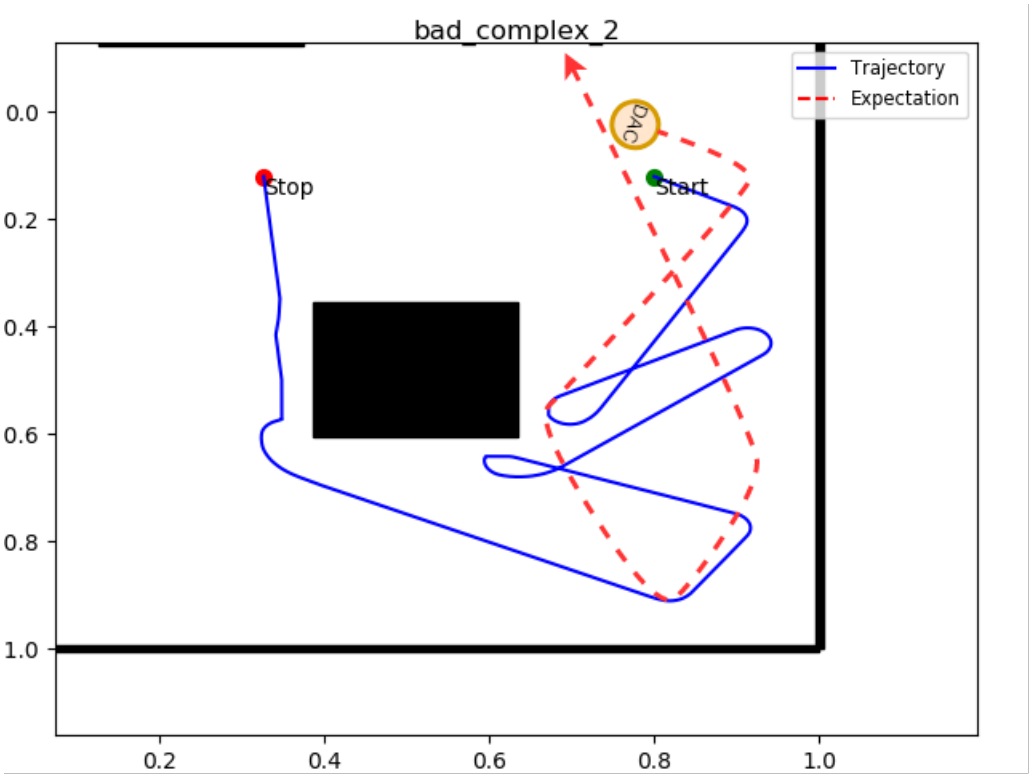


(a) Traiettoria

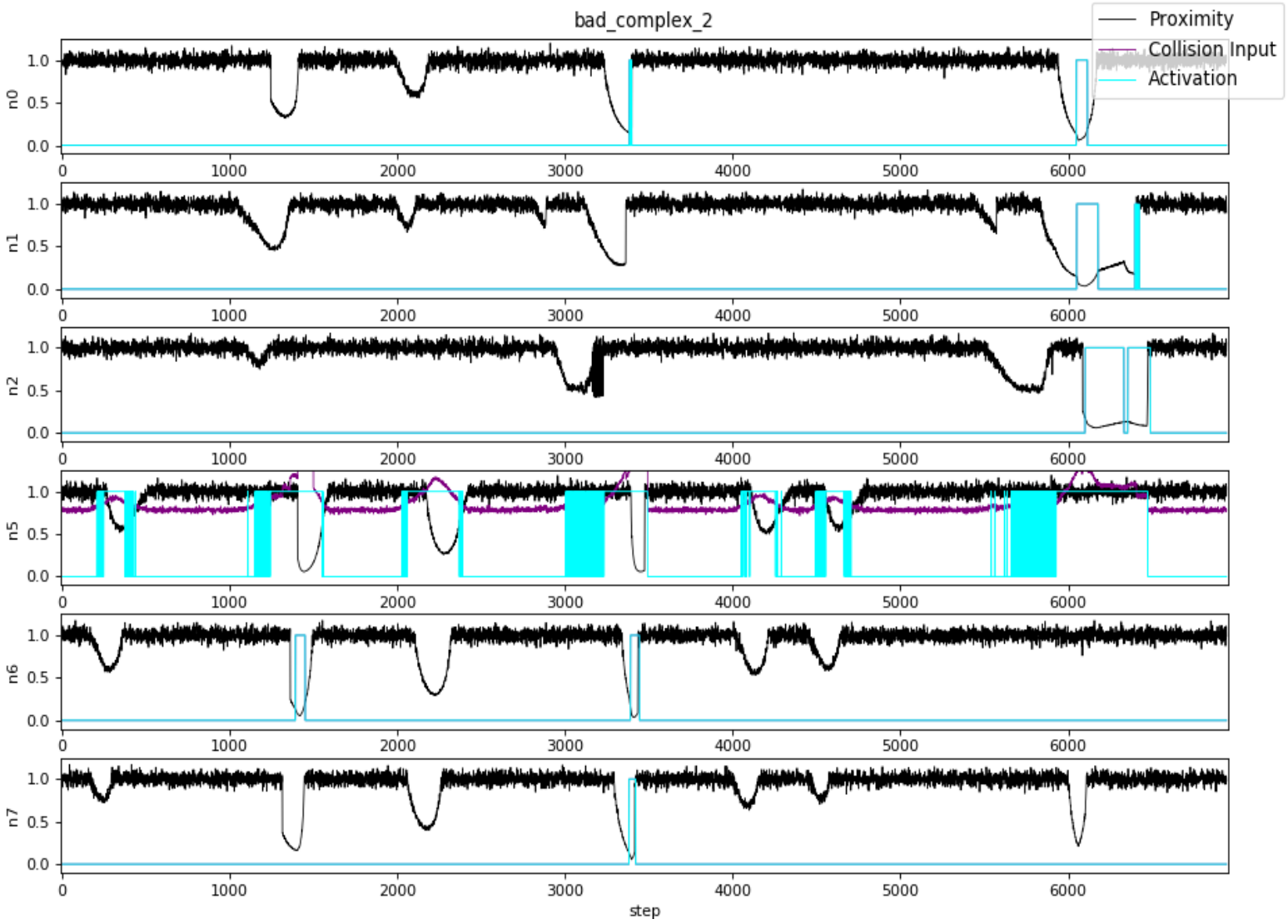


(b) "Encefalogramma"

Figura 17: Ostacolo Complesso



(a) Traiettoria



(b) "Encefalogramma"

Figura 18: Ostacolo Complesso

4 Apprendimento

Nella seguente sezione verrà esplorata l'evoluzione della rete neurale nel corso del processo di apprendimento. Si propone di seguito, in Figura 19, una serie di grafici rappresentanti la crescita dei pesi che connettono i neuroni del layer di Prossimità a quelli del layer di Collisione.

Ogni grafico è riferito ad un neurone del layer di Collisione (specificato sull'asse delle y) mentre il plot è relativo al valore del peso massimo, del peso minimo e della media di tutti i pesi che connettono il neurone.

Com'è possibile vedere in Figura 19, le uniche connessioni che vedono degli incrementi significativi del loro peso sono quelle relative ai neuroni associati al lato sinistro dell'agente. Malgrado ciò, le sole a mantenere un valore costante e non triviale sono, guarda caso, quelle relative al neurone `5`.
Tutte le altre connessioni vedono una decisa oscillazione tra valori prossimi allo zero (se non del tutto nulli) e sporadici picchi in corrispondenza delle sterzate. Tali picchi, però, non sono necessariamente associabili alle sole sterzate dovute a collisioni, bensì potrebbero essere causate anche da una corretta evitazione dell'ostacolo. Infatti, la *learning function* adottata

$$\Delta w_{ij} = (\eta \cdot o_i \cdot o_j - \epsilon \cdot \overline{o_j} \cdot w_{ij})/N_{prox}$$

consente al modello di apprendere solo in corrispondenza di neuroni del layer di Collisione che presentano un'attivazione $o_j = 1$. Ciò implica che l'apprendimento avviene sia che l'agente colpisca un ostacolo, sia che lo eviti correttamente. D'altra parte, però, questo porta anche a ridurre i pesi per quelle connessioni il cui neurone non è attivo $o_j = 0$ (a meno che il peso w_{ij} sia nullo o tutti i neuroni siano inattivi $\overline{o_j} = 0$).
In secondo luogo, il fatto che solo i neuroni del lato sinistro vedano una discreta attività, mentre quelli del lato destro risultano per la maggior parte inerti, è in parte riconducibile a come la manovra d'inversione è stata implementata[8].

Ciò che possiamo vedere in figura potrebbe essere dunque descritto come un fenomeno di Negative Feedback, innescato dalla manovra d'inversione iniziale e potenziato dal meccanismo di apprendimento:

1. L'agente inizia il proprio addestramento volgendo la propria facciata anteriore perpendicolarmente ad un muro dell'arena.

2. Quando l'agente incontra l'ostacolo, Figura 20 – traiettoria `0`, questi inizia a modificare i pesi della rete neurale a partire da quelli relativi ai sensori di

[8] Vedi pagina 2

Collisione frontali. Questo è visibile in Figura 19 come i primi due picchi di crescita dei pesi per i neuroni `0` e `7` del layer di Collisione. Dal momento che solo questi due neuroni sono attivi, si attiverà il neurone d'inversione che cambierà il senso di rotazione della ruota destra, portando l'agente a ruotare leggermente nella medesima direzione.

3. La rotazione a destra porta all'attivazione dei neuroni laterali di sinistra (in ordine prima `6` e successivamente `5`), in quanto vicini all'ostacolo, incrementando la potenza erogata al motore e conseguentemente l'angolo di sterzata. La fase di sterzata continua fintanto che vengono percepiti ostacoli e rallenterà quanto più l'agente si allontana dall'ostacolo, a causa (e causando) la disattivazione progressiva dei neuroni dei sensori più lontani. Dal momento che gli altri neuroni sono inattivi, il valore del loro peso inizia a diminuire velocemente dopo il picco inizialmente registrato.

4. Successivamente, avendo l'agente svoltato a destra, si continuerà ad incontrare ostacoli lungo il lato sinistro (a causa della configurazione dell'arena). Ciò porta ad un continuo e maggiore rafforzamento delle connessioni del lato sinistro e nello specifico del neurone `5`, in quanto è quello che permane attivo per più step (vedi Figura 21) e di conseguenza subirà in quantità molto minore l'effetto di *Active Forgetting*.

5. Nel momento in cui l'agente incontra un ostacolo sul lato destro, Figura 20 – traiettoria `13`, il peso delle connessioni è tale da attivare il neurone `5` sebbene in posizione opposta (portando in questo caso l'agente a colpire l'ostacolo). Questo, come menzionato precedentemente, è dovuto al fatto che tutti i neuroni ricevono i feedback dei valori di prossimità anche dai sensori del lato opposto; perciò se i pesi di un neurone sono abbastanza grandi vi è la possibilità che quel neurone si attivi sebbene l'ostacolo sia sul lato opposto.

   Ciò comporta due importanti conseguenze:

   (1) Impedisce ai pesi dei neuroni di destra di incrementare il peso delle connessione entranti, in quanto non toccheranno mai gli ostacoli

   (2) Tutti i neuroni inattivi vedono i propri pesi decrementati proporzionalmente al valore del peso precedente (già piccolo) e al valore medio di attivazione complessiva del layer di Collisione ($\overline{o_j} = 1/6$ in quanto solo un neurone si attiva nel nostro caso).

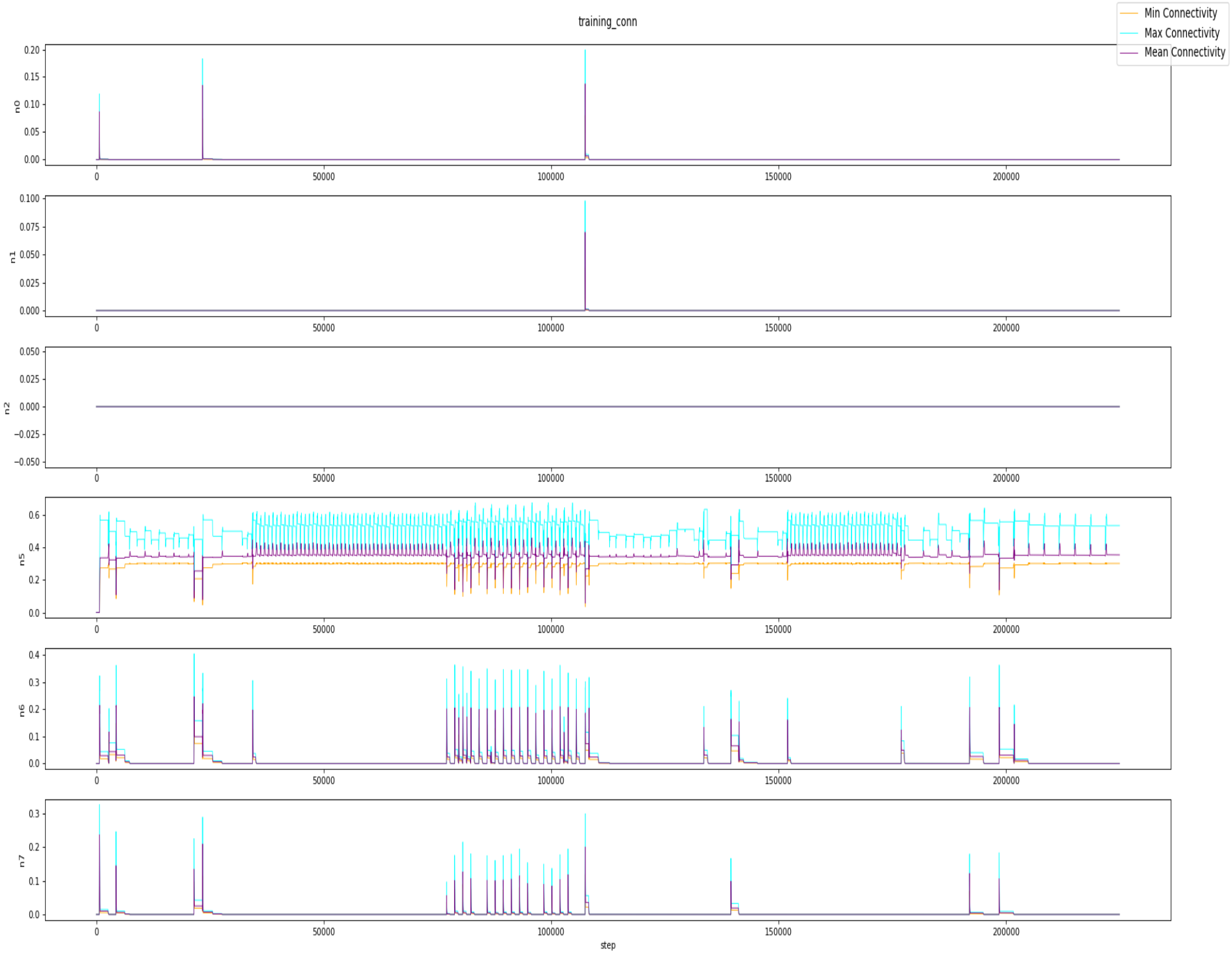

training_conn
Min Connectivity
Max Connectivity
Mean Connectivity
n0
n1
n2
n5
n6
n7
step

Figura 19

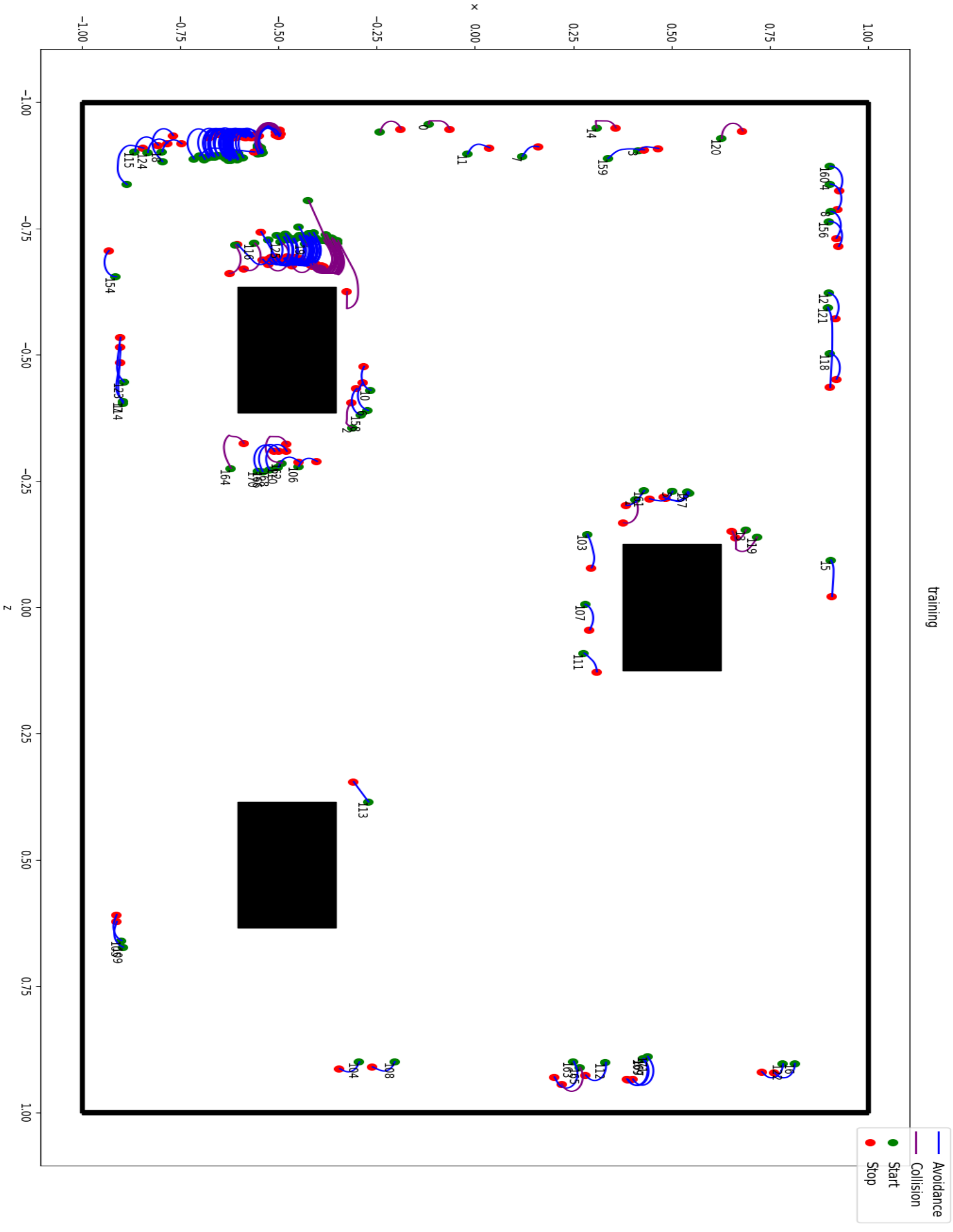


Figura 20

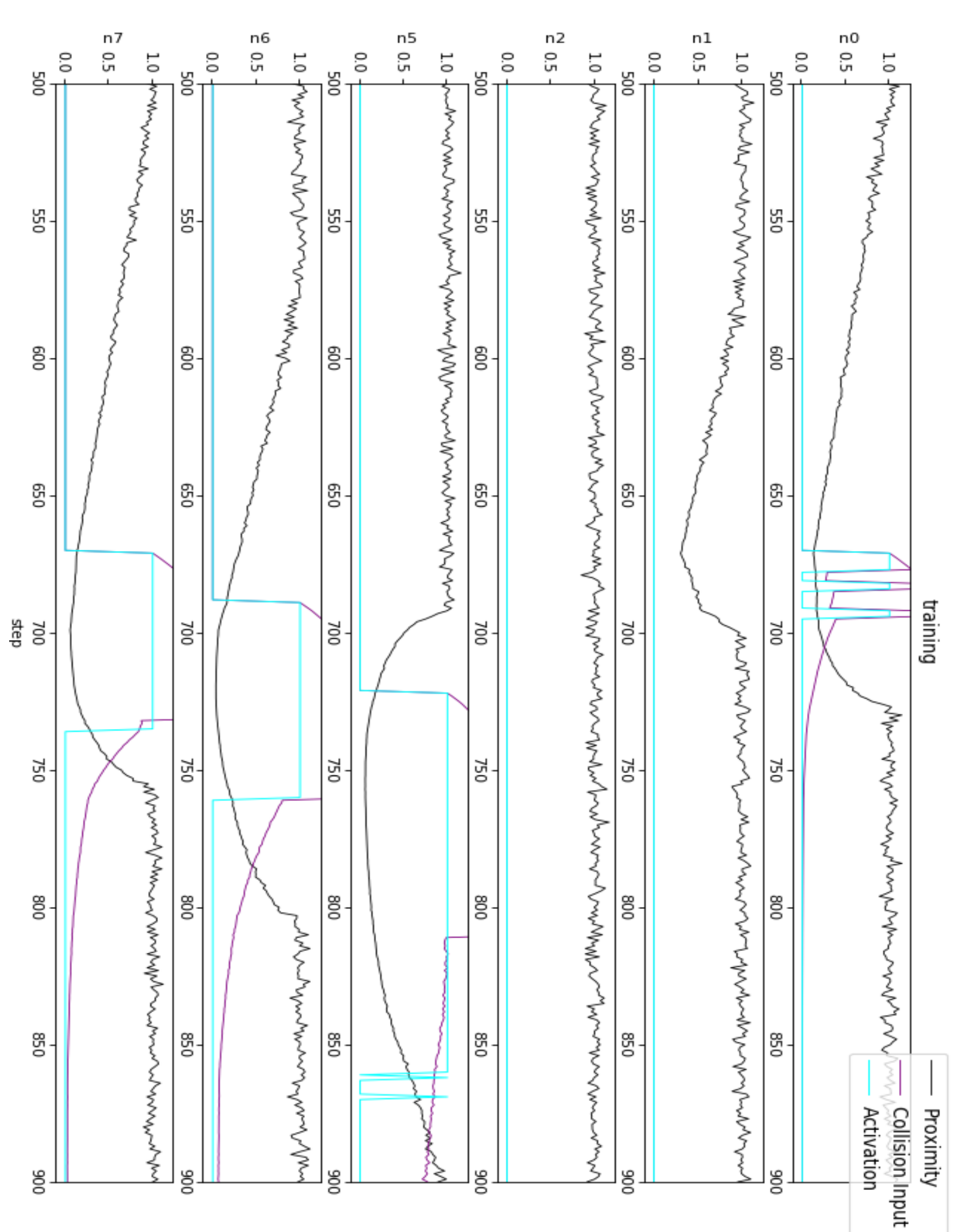


Figura 21: Zoom Addestramento – Attivazione inversione di marcia

5 Conclusioni

In conclusione possiamo dire che, sebbene l'agente abbia dimostrato di aver appreso di evitare gli ostacoli, il comportamento appreso non coincide affatto con il comportamento atteso, ovvero anticipare il comportamento assunto dall'agente quando questi *colpisce* gli ostacoli. Sebbene completamente differenti tra loro, il comportamento appreso dimostra pattern di attivazione inattesi ma funzionali allo svolgimento del suo compito (totalizzando un punteggio molto alto secondo la metrica specificata nella sezione conclusiva di [1]), malgrado all'occhio umano alcune "scelte" del robot risultino inadeguate e poco intelligenti.

Alcune migliorie effettuabili sul modello, sulla base degli elementi di "disturbo" constatati nelle pagine precedenti, potrebbero essere:

- Modificare la struttura della rete neurale rimuovendo lo strato fully-connected e sostituendolo con uno solo parzialmente connesso, in modo tale che i neuroni dei sensori di prossimità del lato sinistro saranno associati solo ai neuroni di collisione del medesimo lato; viceversa per il lato destro. In questo modo dovrebbe essere possibile limitare le attivazioni spurie dovute a pesi troppo grandi per ostacoli sui lati opposti.
- Limitare l'apprendimento alle sole collisioni.
- Dal momento che il comportamento dell'agente deve essere simmetrico potrebbe risultare conveniente implementare un apprendimento di tipo simmetrico, in modo tale che i neuroni opposti possiedano connessioni con pesi uguali.
- Realizzare uno layer di controllo (Motor e Reverse Layer) con azioni più complesse, specie nel caso dell'azione di inversione di marcia, in modo tale da alternare inversioni a destra e a sinistra e dunque sia possibile stimolare equamente i neuroni di ambo i lati.

Tali migliorie, però, non necessariamente porteranno l'agente a far coincidere il comportamento atteso da quello appreso. Sicuramente quelle che porterebbero i migliori risultati senza compromettere l'idea iniziale sono la numero `1` e `4`. La prima permette di ridurre la complessità della rete mantenendo comunque un certo grado di elasticità, la seconda, invece permetterebbe un maggior controllo sulle azioni del robot e permetterebbe di rendere il sistema più che semplicemente reattivo.

Riferimenti bibliografici

[1] Gabellini Cevoli. Riproduzione di un distributed adaptive controller per agenti robotici: Apprendimento autonomo della obstacle-avoidance, 2019. Available at https://github.com/XanderC94/IRSProject/releases/download/0.1/Relazione_IRS.pdf.

[2] Olivier Michel. Cyberbotics ltd. webots™: professional mobile robot simulation. *International Journal of Advanced Robotic Systems*, 1(1):5, 2004.

[3] Rolf Pfeifer and Christian Scheier. *Understanding intelligence*. MIT press, 2001.

[4] Paul FMJ Verschure, Ben JA Kröse, and Rolf Pfeifer. Distributed adaptive control: The self-organization of structured behavior. *Robotics and Autonomous Systems*, 9(3):181–196, 1992.