

UNIVERSITÀ DEGLI STUDI DI GENOVA

CORSO DI STUDIO IN INGEGNERIA BIOMEDICA

Relazione finale per il conseguimento del titolo di
Dottore in Ingegneria Biomedica

Implementazione di una Rete di Neuroni “Spiking” per il Controllo di Robot Quadrupedi

Manuel Delucchi

Febbraio 2023

Relatore: Prof. Marco Storace

Correlatore: Dott.ssa Valentina Baruzzi

Sommario

La biorobotica può essere definita come la scienza e tecnologia della progettazione e della realizzazione di sistemi robotici di ispirazione biologica. Gli obiettivi che si pone sono:

- analizzare e studiare come funzionano i sistemi biologici e usare questi modelli per progettare nuovi e migliori sistemi robotici che imitano le funzionalità della loro controparte biologica;
- sviluppare piattaforme fisiche equivalenti ai sistemi biologici, al fine di validare sperimentalmente “modelli” di sistemi viventi e i loro principi funzionali.

Questa tesi si lega al primo obiettivo e tratta, in particolare, della locomozione di un robot quadrupede. Il controllo della locomozione è un problema complesso nel campo della robotica, nonostante appaia piuttosto semplice in natura. Imparare ad adattare la propria andatura ai cambiamenti ambientali gioca un ruolo essenziale nella locomozione dei robot dotati di gambe.

Come primo passo verso questo obiettivo, in questa tesi viene proposto l’uso di metodologie bio-ispirate per il controllo della locomozione di un cane robotico. In particolare, viene utilizzato un modello di Central Pattern Generator, ovvero una piccola rete di neuroni capace di produrre autonomamente attività ritmica, che biologicamente ha un ruolo fondamentale nella locomozione animale. Questa rete viene addestrata, attraverso un algoritmo di apprendimento stocastico, a riprodurre cinque diverse andature tipiche dei quadrupedi.

Inoltre, si è sviluppato un controllore robotico che incorpora la rete addestrata e oscillatori dinamici per la generazione delle traiettorie delle zampe del robot quadrupede. L’efficacia della strategia di controllo proposta è verificata nell’ambiente di simulazione Webots, tramite la versione virtuale del cane robotico Bioloid.

Parole chiave: Central Pattern Generators (CPG), Locomozione, Quadrupede

1 Introduzione

I Central Pattern Generators (CPG) sono reti neurali relativamente piccole presenti sia nei vertebrati sia negli invertebrati [5]. In particolar modo, nei vertebrati, sono poste a livello del midollo spinale. La funzione principale che rivestono riguarda il controllo di movimenti ripetitivi, come la locomozione, la respirazione e la masticazione, mediante la generazione di pattern ritmici per l'attivazione dei muscoli. I CPG possono svolgere autonomamente la propria funzione anche in assenza di feedback sensoriali provenienti dal sistema nervoso periferico (SNP) [7, 5] o da segnali provenienti dal sistema nervoso centrale (SNC). Tuttavia, alcune aree superiori come la corteccia motoria, il cervelletto e i gangli della base possono modulare l'azione dei CPG in base a informazioni relative all'ambiente circostante [1, 6].

Con il termine “locomozione” si intende la capacità di un animale di spostarsi da un punto A a un punto B; la locomozione è basata su movimenti ritmici ed alternati di tutto il corpo e delle sue appendici, che devono potersi adattare istante per istante alle condizioni ambientali. La contrazione e il rilassamento ritmico dei muscoli, regolati dai CPG, azionano gli arti in un ordine specifico, generando le diverse andature che si osservano nella locomozione animale [1]. Compiti quali evitare ostacoli o raggiungere una preda richiedono la locomozione in una particolare direzione e alla velocità desiderata [1]. I CPG devono quindi essere in grado di generare andature multiple, passando da un'andatura all'altra e regolandone la velocità in base a segnali di controllo provenienti dai centri cerebrali superiori.

Lo studio dei meccanismi che regolano la locomozione animale trova applicazione anche nello sviluppo di controllori bio-ispirati per la locomozione robotica. In questo ambito, è di fondamentale importanza lo sviluppo di modelli matematici di CPG. Gli obiettivi principali di questa tesi sono: (i) modellare una rete CPG che regoli la locomozione quadrupede, in grado di generare cinque andature (camminata laterale, camminata diagonale, galoppo, saltello e trotto), (ii) realizzare un algoritmo di apprendimento supervisionato per ottimizzare i pesi della rete CPG, (iii) sviluppare un controllore per robot quadrupedi sulla base del modello di CPG e (iv) verificare la sua efficacia in un ambiente di simulazione robotica.

2 Stato dell'arte

L'interazione tra biologia e robotica può svilupparsi in due direzioni: la robotica prende ispirazione da meccanismi biologici per sviluppare tecnologie di controllo innovative, ma, allo stesso tempo, un numero sempre maggiore di progetti cerca di cambiare prospettiva, usando robot come strumenti scientifici per validare modelli biologici. Lo sviluppo di un controllo della locomozione di robot basato su CPG è interessante non solo per la comprensione dei sistemi di locomozione animale, ma anche perché può fornire un controllo agile, robusto ed efficiente dal punto di vista energetico durante il movimento in ambienti complessi [6].

Gli studi condotti sui CPG hanno portato alla definizione di diversi modelli, alcuni dei quali hanno trovato applicazione nell'ambito del controllo robotico grazie alle loro particolari proprietà. Per esempio, alcuni di questi sono stati utilizzati con robot esapodi che si ispirano alla locomozione degli insetti [3]. Ci si serve dei CPG anche per controllare la locomozione di robot che nuotano; ad esempio, nella nuotata della lampreda, un pesce primitivo simile all'anguilla, il corpo del pesce partecipa al movimento in tutta la sua lunghezza attraverso un susseguirsi di ondulazioni a cui l'acqua oppone resistenza [5, 6, 7]. Grazie alla sua relativa semplicità, la lampreda è il pesce che è stato maggiormente modellato come robot.

Particolare attenzione è stata rivolta allo studio del controllo del movimento dei quadrupedi con l'utilizzo di CPG. Un esempio riguarda il controllo della locomozione del robot quadrupede HyQ, creato e studiato dall'IIT (Istituto Italiano di Tecnologia). Il robot è in grado di riprodurre diverse andature e movimenti, sia altamente dinamici come la corsa e il salto, sia prudenti come una camminata su terreni accidentati [3]. Un altro esempio è AIBO, robot quadrupede del quale viene studiata la locomozione in [2]. Tra le altre cose, sono stati esplorati diversi mezzi per l'integrazione del feedback sensoriale nel controllo della locomozione, ed è stato scoperto che il feedback sensoriale che modula l'attività del CPG tende a garantire una locomozione più stabile su terreni complessi, al contrario del feedback indipendente dall'attività del CPG [5]. Per i quadrupedi, il lavoro è più difficoltoso data la complessità dell'azione neuronale: oltre a controllare l'attività dei muscoli flessori ed estensori, vanno coordinati anche i movimenti delle zampe in senso antero-posteriore e in modo bilaterale. Per questo

motivo, i modelli riguardanti le andature di quadrupedi sono ancora a un livello piuttosto astratto [5]. Similmente alla lampreda, vi sono altri robot che incorporano modelli di CPG. Per esempio, la salamandra, un anfibio in grado di nuotare e camminare, offre un interessante collegamento tra la ricerca sulla lampreda e quella sui quadrupedi [6]. Modelli di CPG vengono impiegati sempre più ampiamente anche in robot umanoidi [5]. Un aspetto interessante dell’uso dei robot nelle neuroscienze computazionali è che essi consentono di validare i modelli computazionali in un ambiente reale. In particolare, ciò significa che i modelli possono essere testati all’interno di un ciclo completo di rilevamento sensoriale e azione. Questo è importante, perché alcuni aspetti della locomozione possono dipendere in modo critico dall’interazione con l’ambiente [5]. I simulatori dinamici possono essere utilizzati per simulare la fisica del corpo e dell’ambiente, ma bisogna rendersi conto che forniscono solo una prima approssimazione. Alcune forze di interazione, come le forze di contatto, le forze di attrito e le forze idrodinamiche sono estremamente difficili da simulare correttamente, soprattutto per corpi provvisti di arti che si muovono e cambiano assetto [5, 6]. Infine, l’uso dei robot costringe a puntare a una comprensione completa del funzionamento di un sistema in grado di muoversi autonomamente. Il “fallimento” è molto visibile in un robot (ad esempio, cade o si blocca), e il sistema di controllo deve garantire che il robot funzioni correttamente. Per esempio, nel caso della locomozione, questo richiede di risolvere correttamente i problemi di generazione del ritmo, coordinazione tra i gradi di libertà, controllo dell’equilibrio, della velocità e della direzione. Non bisogna tuttavia sottovalutare le difficoltà nell’utilizzo di robot per le neuroscienze computazionali. Prima di tutto, è molto difficile, se non impossibile, replicare correttamente le proprietà biomeccaniche dei corpi animali, in particolare il loro numero di gradi di libertà, la distribuzione della massa e le proprietà viscoelastiche. Analogamente, mentre alcune modalità sensoriali possono approssimare correttamente quelle biologiche, come la visione e l’elaborazione del suono, altre, come il tatto e la propriocezione, sono ancora lontane dall’essere replicate correttamente dalle attuali tecnologie dei sensori, anche se, negli ultimi anni, la ricerca sta facendo passi da gigante.

3 Materiali e metodi

Il CPG considerato in questa tesi è stato sviluppato sulla base della rete proposta in [1]; in particolare, la rete usata in [1] per il controllo di un robot esapode è stata riformulata per il controllo della locomozione di un robot quadrupede, come spiegato nel seguito. Il modello di CPG è costituito da 4 neuroni “spiking” (N_1, N_2, N_3 e N_4) connessi *all-to-all*, ciascuno corrispondente ad un arto del robot quadrupede. Nel controllore robotico, lo “spike” di un neurone del CPG provoca il movimento dell’arto robotico corrispondente. Pertanto, ogni andatura corrisponde ad una specifica sequenza di “spike” dei neuroni della rete.

Il primo obiettivo della tesi è ottenere un singolo CPG in grado di riprodurre più andature, ognuna delle quali può essere attivata da un diverso “neurone di controllo”, che imita l’azione delle aree cerebrali superiori. Le cinque andature di riferimento sono: camminata laterale, camminata diagonale, galoppo, trotto e saltello. Di conseguenza, la rete comprende 5 neuroni “spiking” di controllo (G_1, G_2, \dots, G_5) per la selezione dell’andatura. Il modello completo è denominato SCPG (Spiking CPG). Ciascun neurone è descritto da un modello Leaky Integrate and Fire (LIF) discretizzato [3], largamente utilizzato per modellare neuroni spiking. Il potenziale di membrana (V_j) è espresso come:

$$V_j[t+1] = \frac{V_j[t]}{\alpha} + \sum_i W_{ij} S_i[t] \quad (1)$$

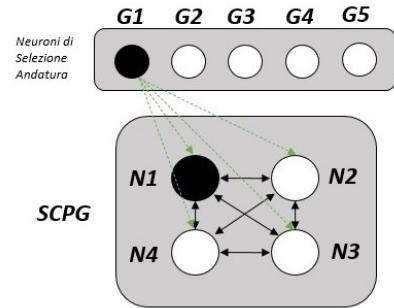


Figura 1: “SCPG Network”. N_1, N_2, N_3 e N_4 rappresentano rispettivamente la gamba anteriore sinistra, anteriore destra, posteriore destra e posteriore sinistra di un robot quadrupede.

$$\text{se } V_j[t] > V_{th} \text{ allora } S_j[t+1] = 1, V_j[t+1] = 0 \quad (2)$$

La corrente di perdita è modellata con un fattore di decadimento α (α deve essere > 1 per essere fisicamente ragionevole, in questa tesi $\alpha = 1.1$). Quando il potenziale di membrana V_j supera la soglia di spike V_{th} , viene emesso uno spike S_j e il potenziale di membrana viene istantaneamente riportato al potenziale di riposo, in questo caso fissato a zero (2). Il secondo termine della (1) corrisponde alla corrente sinaptica. Uno spike pre-sinaptico, S_i , determina l'incremento del potenziale di membrana dei neuroni post-sinaptici. I coefficienti sinaptici (W_{ij}) pesano opportunamente gli ingressi dai neuroni presinaptici (i-esimo neurone) al neurone postsinaptico (j-esimo neurone), come mostrato nell'equazione.

Le sequenze di spike corrispondenti alle 5 andature di riferimento sono elencate in Tab. 1. Ad esempio, quando il neurone di selezione associato alla camminata laterale si attiva, N_1 deve attivarsi nel successivo istante temporale (t_1), N_4 in quello dopo (t_2), e così via, ripetendo ciclicamente la sequenza di attivazione. Pertanto, i pesi W_{ij} delle connessioni della rete devono essere programmati in modo tale che, quando un neurone G_i di selezione dell'andatura si attiva, i neuroni N_1, \dots, N_4 si attivino negli istanti temporali successivi con la sequenza corrispondente all'andatura desiderata.

	C. Laterale				C. Diagonale				Galoppo				Trotto		Saltello	
	1	0	0	0	1	0	0	0	1	0	0	0	1	0	1	0
N1	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0
N2	0	0	1	0	0	0	0	1	0	0	0	0	0	1	1	0
N3	0	0	0	1	0	1	0	0	0	1	0	0	1	0	0	1
N4	0	1	0	0	0	0	0	1	0	0	0	1	0	1	0	1
	t₁	t₂	t₃	t₄	t₁	t₂	t₃	t₄	t₁	t₂	t₃	t₄	t₁	t₂	t₁	t₂

Tabella 1: Pattern delle cinque andature, in ordine (da sinistra a destra): Camminata Laterale, Camminata Diagonale, Galoppo, Trotto e Saltello.

Allo scopo di programmare i pesi W_{ij} delle connessioni della rete, è stato implementato su Matlab l'algoritmo stocastico di apprendimento supervisionato descritto in [1], opportunamente adattato per la rete e le andature considerate. Inizialmente, tutti i pesi W_{ij} sono inizializzati in modo casuale. L'algoritmo di learning modifica i pesi allo scopo di rendere i pattern di spike generati dalla rete (a seguito dello spike di un neurone di controllo) identici ai pattern di spike richiesti (vedi Tab. 1). Per allinearsi alla sequenza di spike richiesta, i neuroni che sparano in modo ridondante sono “soppressi”, diminuendo in modo stocastico i pesi sinaptici delle loro connessioni afferenti, e quelli che non sparano quando richiesto sono “eccitati”, aumentando analogamente i pesi sinaptici delle connessioni afferenti. Questa procedura di aggiornamento è ripetuta per ogni andatura. Ad ogni istante della sequenza, il numero di spike errati costituisce l'errore della rete. L'algoritmo di learning arriva a convergenza, determinando la configurazione definitiva dei pesi W_{ij} , quando l'errore accumulato in tutti gli istanti temporali, per tutte le andature, è nullo. La convergenza dell'algoritmo di learning indica la corretta programmazione di tutte le andature nel modello di SCPG.

Il modello di SCPG con i pesi delle connessioni correttamente programmati può essere utilizzato per gestire l'alternanza delle gambe in un controllore robotico. In questa tesi, è stato sviluppato un controllore per la locomozione del robot quadrupede Bioloid Dog (Fig. 2), utilizzando il linguaggio di programmazione C++. Il corretto funzionamento del controllore è stato verificato tramite l'ambiente di simulazione 3D open-source Webots.

In [4], i ricercatori dell'Istituto Italiano di Tecnologia (IIT) hanno proposto un modello di oscillatore che fornisce traiettorie ellittiche per le zampe di un robot quadrupede (Fig. 2). Queste sono generate da quattro oscillatori non lineari, il cui stato rappresenta le coordinate cartesiane di ciascuna zampa. In [4] gli oscillatori sono connessi tra loro e diverse andature possono essere ottenute modificando tali connessioni. In questa tesi, invece, gli oscillatori sono disaccoppiati e ognuno è comandato dal neurone corrispondente della rete SCPG: in questo modo tutte le andature sono ottenute da un'unica rete semplicemente attivando il neurone di selezione dell'andatura desiderata (le equazioni relative agli oscillatori sono descritte in [4]). Il sistema di oscillatori scelto per la generazione delle traiettorie può essere esteso in modo da adattare dinamicamente la forma del passo alle irregolarità del terreno

sulla base del feedback sensoriale, come già dimostrato in [4]. Tale aspetto non è stato trattato in questa tesi e si presta a sviluppi futuri di questo progetto.

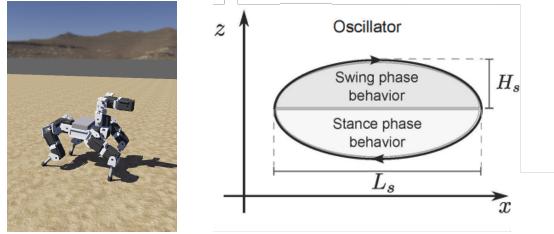


Figura 2: Bioloid Dog - Webots Simulation Environment (a sinistra). Traiettoria del piede generata dall’oscillatore CPG (a destra). L_s e H_s sono rispettivamente la lunghezza e l’altezza di un singolo passo.

Muovere le zampe del robot in modo che percorrano la traiettoria desiderata richiede l’attivazione sinergica dei diversi motori di ogni arto robotico. Ogni arto robotico è formato da una serie di segmenti rigidi connessi tra di loro per mezzo di articolazioni (giunti), perciò, data la traiettoria di ogni zampa, è possibile calcolare gli angoli ai giunti usando la cinematica inversa, ossia le seguenti equazioni:

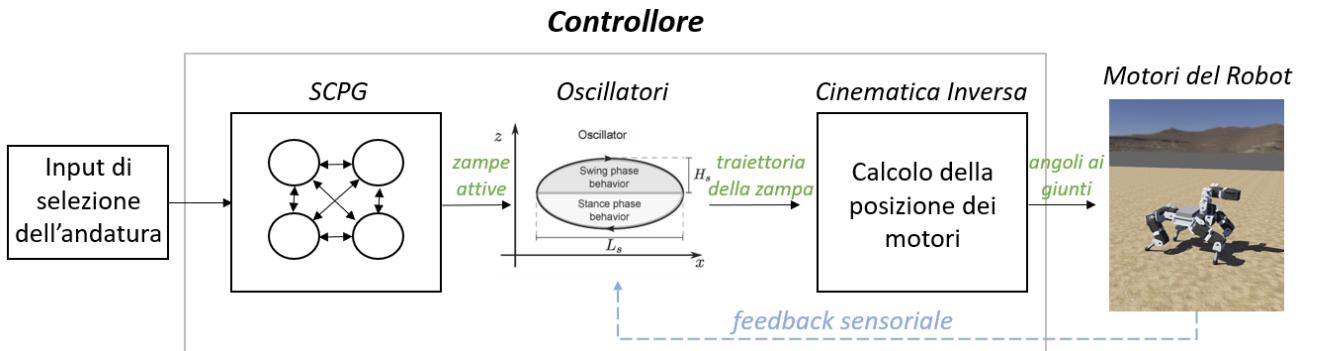
$$\theta_2 = \arccos\left(\frac{x^2 + z^2 - L_1^2 - L_2^2}{2L_1L_2}\right) \quad (3)$$

$$\theta_1 = \arccos\left(\frac{(L_1 + L_2\cos(\theta_2))(-x) - (-L_2\sin(\theta_2))z}{(L_1 + L_2\cos(\theta_2))^2 + (-L_2\sin(\theta_2))^2}\right) \quad (4)$$

dove x e z sono le coordinate cartesiane delle zampe, mentre L_1 e L_2 corrispondono, rispettivamente, alla lunghezza dell’estremità superiore e di quella inferiore della gamba.

θ_1 e θ_2 , invece, sono, rispettivamente, gli angoli al giunto dianca e di ginocchio, usati per controllare i motori. In figura 3 viene illustrata l’azione del controllore completo: la rete SCPG riceve in input un comando di selezione dell’andatura, convertito nello spike di uno dei neuroni di controllo. Ad ogni istante, l’attività della rete determinerà quali zampe dovranno attivarsi. Tale informazione viene prima convertita nella traiettoria ellittica di ogni zampa dall’oscillatore, e poi negli angoli ai giunti per mezzo della cinematica inversa. I motori del robot ricevono in input, istante per istante, il valore dell’angolo che ogni giunto deve assumere. La freccia di retroazione tratteggiata rappresenta la possibilità di sfruttare il feedback sensoriale (raccolto dai sensori di posizione del robot) per modificare dinamicamente (in tempo reale) la traiettoria di ogni zampa, andando quindi a correggere eventuali errori di traiettoria e migliorare la stabilità del suo andamento, in particolare su terreni irregolari. Da sottolineare che l’uscita della rete SCPG viene ricalcolata ogni $T' = T/2$, dove T è il periodo degli oscillatori. Questo garantisce un movimento fluido del robot ed evita pause eccessive tra il movimento di un arto e del successivo.

Figura 3: Schema a Blocchi del Controllore



4 Risultati

La prima parte di questa sezione analizza la capacità della rete SCPG di memorizzare più andature tramite l'algoritmo di apprendimento stocastico. Il risultato ottenuto è mostrato in Figura 4: i rettangoli bianchi indicano gli spike dei neuroni corrispondenti, mentre i rettangoli neri indicano l'assenza di attività. Confrontando il grafico con Tab. 1, si può osservare come la sequenza di spike sia corretta per tutte e 5 le andature, confermando come queste siano state programmate correttamente nella rete. Si consideri, per esempio, il trotto: nell'istante iniziale N_1 ed N_3 sono attivi mentre nell'istante successivo sono attivi N_2 ed N_4 ; in figura, questa sequenza è ripetuta per 10 volte, corrispondente al numero di ripetizioni usate per ognuna delle andature nel processo di apprendimento. Dopo aver ricevuto un segnale di controllo per la selezione dell'andatura, tradotto nello spike di uno dei neuroni di controllo G_1, \dots, G_5 , la rete è in grado di riprodurre la sequenza di spike corrispondente indefinitamente, fino alla ricezione di un altro segnale di controllo.

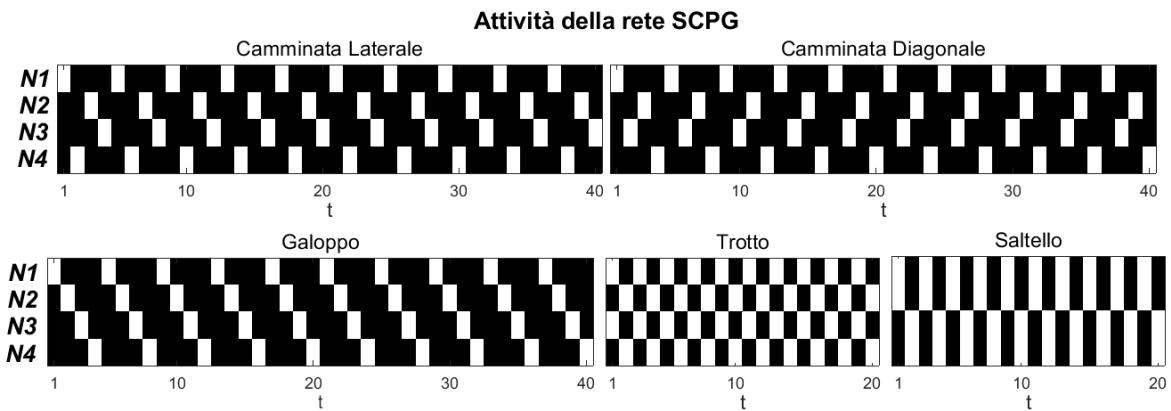


Figura 4: Attività della rete SCPG. I rettangoli neri corrispondono a 0 (No Spike), mentre quelli bianchi ad 1 (Spike) - prendendo come riferimento la Tab. 1 introdotta nella sezione precedente.

Si sottolinea che tutte le andature sono ottenute dalla stessa rete SCPG, i cui pesi W_{ij} sono stati ricavati tramite l'algoritmo di apprendimento. A causa della stocasticità dell'algoritmo, non tutte le procedure di ottimizzazione raggiungono la convergenza (errore nullo). Inoltre, all'aumentare del numero di andature da memorizzare, la percentuale di convergenze diminuisce drasticamente. Pertanto, è stata eseguita un'analisi sulla percentuale di convergenze al variare del numero di andature da memorizzare, lanciando $n_{trials} = 100$ volte la procedura di ottimizzazione per ciascun caso. E' stato osservato che è possibile apprendere tre andature (in particolare camminata, trotto e saltello) con un tasso di convergenza maggiore dell'80%. Aggiungendo un'ulteriore andatura, la camminata diagonale, il tasso di convergenza subisce una drastica riduzione, arrivando a circa il 20%. Infine, aggiungendo il galoppo come quinta andatura, il risultato ottenuto è un tasso di convergenza pari all'1%. La rete è quindi in grado garantire la possibilità di memorizzare fino a cinque andature uniche ma, la bassa percentuale di convergenza, rende l'algoritmo di ottimizzazione poco "robusto" dato che, solo in pochi casi, trova una configurazione di pesi accettabile.

La rete SCPG con i pesi W_{ij} ottenuti dall'algoritmo di apprendimento, in grado di generare le cinque andature al variare del controllo, è stata utilizzata all'interno di un controllore robotico. Tale controllore genera le traiettorie ellittiche delle zampe del robot nel piano $x - z$ come variabili di stato di quattro oscillatori, uno per ogni zampa. I parametri delle equazioni differenziali che governano la dinamica degli oscillatori sono stati impostati in modo da ottenere traiettorie ellittiche con un asse orizzontale (in x) di 10 cm e un asse verticale (in z) di 1,74 cm, in accordo con le dimensioni e i limiti fisici del robot. Il periodo degli oscillatori, ossia il tempo impiegato per completare un passo, è $T = 2.528$ s. La posizione delle zampe nel piano cartesiano $x - z$ è tradotta in angoli ai giunti dalla cinematica inversa e inviata agli attuatori del robot per controllarne il movimento. Figura 5 mostra l'andamento delle posizioni x e z di ciascuna zampa del robot generate dal controllore. Si sottolinea che la variabile x rappresenta lo spostamento orizzontale di ciascuna zampa nel tempo, mentre la variabile z rappresenta lo spostamento verticale.

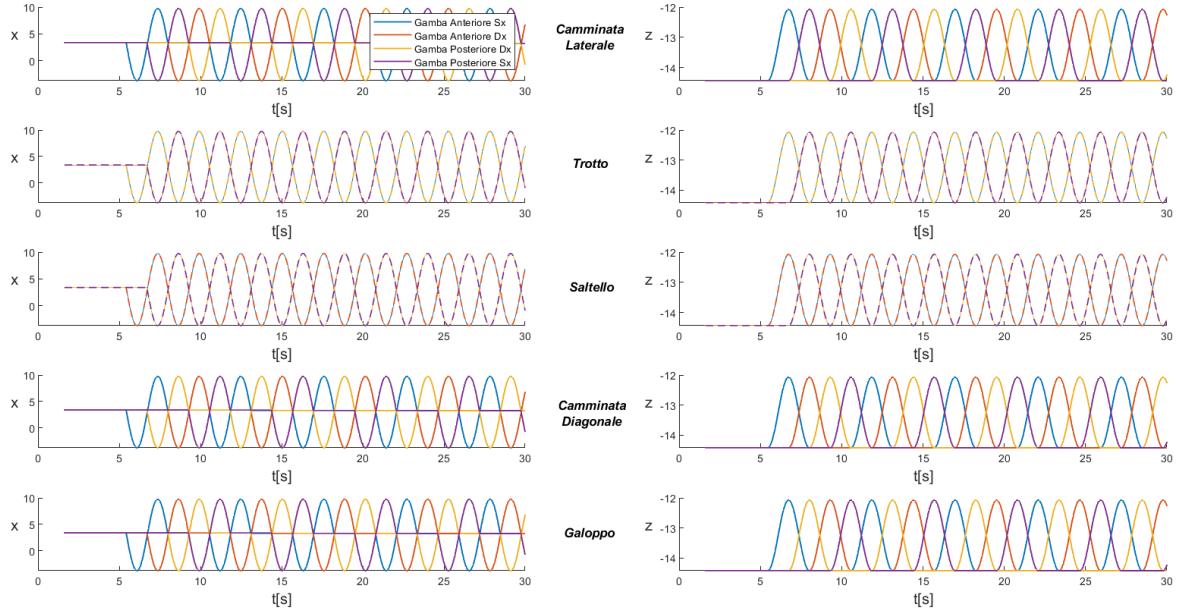


Figura 5: Grafici delle posizioni x e z nel tempo generate dal controllore per ciascuna zampa del robot. A sinistra l’andamento della x nel tempo, a destra l’andamento della z nel tempo. Si noti che il punto più basso dell’ellisse corrisponde alla coppia di coordinate $x_0 = 3.39509$ e $z_0 = -14.4202$. Le tracce sono state ottenute mediante una simulazione della durata di $t = 30$ s per ciascuna andatura.

Analizzando, per esempio, le posizioni x e z nel tempo relative alla camminata diagonale, si osserva che le gambe che si attivano sono, in ordine: l’anteriore sinistra (blu), la posteriore destra (giallo), l’anteriore destra (rosso) e la posteriore sinistra (viola). La traiettoria di ogni passo inizia dal punto (x_0, z_0) , che corrisponde all’estremo inferiore dell’ellisse; come spiegato nella sezione Materiali e Metodi, l’uscita della rete SCPG viene ricalcolata ogni $T' = T/2$. Di conseguenza, a metà del periodo di ciascun passo, si attiva la zampa successiva. In questo modo il periodo di inattività di ciascuna zampa è ridotto al minimo e si ottiene un movimento più fluido. Come si può vedere dai grafici di Fig. 5, le traiettorie delle zampe per tutte le andature risultano simulate correttamente. Per quanto riguarda la camminata laterale e il galoppo, come per la camminata diagonale, l’alternanza delle gambe è chiaramente visibile. Considerando, invece, il trotto e il saltello, dove due zampe vengono attivate contemporaneamente per ciascun istante di tempo, si noti che le sinusoidi relative a queste zampe sono sovrapposte. Il movimento del robot Bioloid è stato simulato tramite il software Webots. La camminata diagonale e il trotto, che richiedono il movimento contemporaneo di zampe posizionate diagonalmente tra loro, sono riprodotte correttamente, come mostrato nei fermo-immagine di Fig. 6. In questi casi il robot mantiene l’equilibrio e avanza. Nel caso della camminata laterale e del galoppo, che richiedono il movimento contemporaneo di zampe ipsilateral, il robot si muove sul posto, senza avanzare. Nel caso del saltello, il robot retrocede. Questo è dovuto al fatto che, nell’ambito di questa tesi, non è stato realizzato il blocco del controllore dedicato al mantenimento dell’equilibrio del tronco del robot, che si occuperebbe di stabilizzarne l’andatura tramite le forze di reazione generate dalle zampe sul terreno.

Lo studio condotto in questa tesi ha quindi permesso di ottenere un singolo CPG, costituito da quattro neuroni, in grado di riprodurre cinque andature tipiche di un quadrupede. Per l’apprendimento di queste è stato proposto un algoritmo basato sull’aggiornamento stocastico dei coefficienti sinaptici delle connessioni della rete. Inoltre, la rete SCPG programmata è stata utilizzata per sviluppare il controllore del robot, sfruttando, in particolare, un sistema di quattro oscillatori non lineari disaccoppiati per la generazione delle traiettorie di ciascuna zampa. Infine, l’efficacia della strategia di controllo proposta è stata verificata nell’ambiente di simulazione Webots.

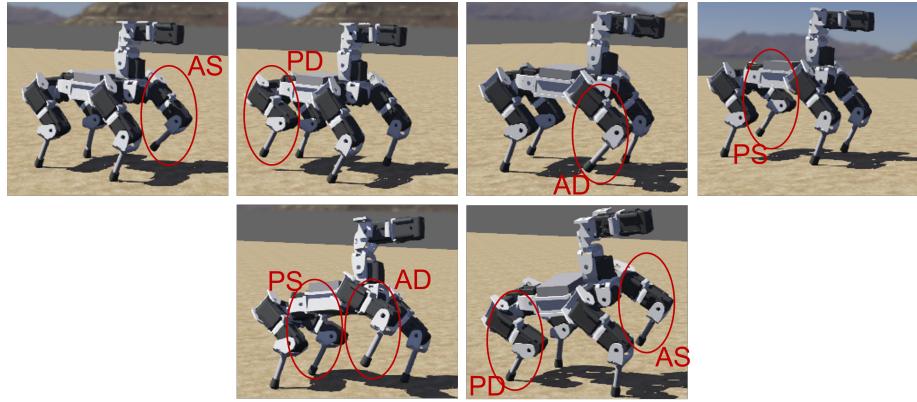


Figura 6: Fermo-immagine ottenuti dalla simulazione Webots del robot. Vengono mostrate le andature camminata laterale (in alto) e trotto (in basso). Le zampe attive in ogni immagine sono cerchiare in rosso (AS = Anteriore Sinistra, PD = Posteriore Destra, AD = Anteriore Destra e PS = Posteriore Sinistra).

Dei potenziali sviluppi di questo lavoro potrebbero essere:

- Aumentare il numero di neuroni della rete: questo da un lato conferisce maggiore flessibilità alla rete, ma aumenta anche le incognite e quindi la complessità del problema di ottimizzazione;
- Sfruttare il feedback sensoriale per modificare dinamicamente la traiettoria di ogni singola zampa in risposta ai cambiamenti dell’ambiente: questo assicura che il robot sia in grado di mantenere stabilità ed equilibrio durante la locomozione;
- Implementare un blocco di controllo del movimento in grado di fornire equilibrio e stabilità al tronco del robot: questo può sfruttare il feedback sensoriale e altre informazioni, come le forze di reazione generate dalle zampe sul terreno, per regolare dinamicamente il movimento del tronco.

Una futura applicazione del lavoro svolto potrebbe essere il suo inserimento nel controllo della locomozione del robot quadrupede HyQ nell’ambito di una collaborazione con l’IIT (Istituto Italiano di Tecnologia).

Riferimenti bibliografici

- [1] A. Lele et al. “An End-to-End Spiking Neural Network Platform for Edge Robotics: From Event-Cameras to Central Pattern Generation”. In: *IEEE Transactions on Cognitive and Developmental Systems* (2022).
- [2] Liu et al. “CPG-Inspired Workspace Trajectory Generation and Adaptive Locomotion Control for Quadruped Robots”. In: *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* (2011).
- [3] V. Barasuol et al. “WCPG: A Central Pattern Generator for Legged Robots Based on Workspace Intentions”. In: *ASME 2011 Dynamic Systems and Control Conference and Bath/ASME Symposium on Fluid Power and Motion Control, DSAC 2011* (gen. 2011).
- [4] Victor Barasuol et al. “A reactive controller framework for quadrupedal locomotion on challenging terrain”. In: *2013 IEEE International Conference on Robotics and Automation* (2013), pp. 2554–2561.
- [5] Auke Jan Ijspeert. “Central pattern generators for locomotion control in animals and robots: A review”. In: *Neural Networks* 21.4 (2008). Robotics and Neuroscience, pp. 642–653.
- [6] Yu.A. Tsybina et al. “Toward biomimetic robotics: A review on swimming central pattern generators”. In: *Chaos, Solitons Fractals* 165 (2022), p. 112864.
- [7] Zhang Yiyang Wang Ming e Junzhi Yu. “An SNN-CPG Hybrid Locomotion Control for Biomimetic Robotic Fish”. In: *Journal of Intelligent Robotic Systems* 105 (giu. 2022).