

COMPLEX DYNAMICS in NON-LINEAR LOCALLY CONNECTED CIRCUITS

REACTION DIFFUSION EQUATIONS

Tramite i concetti qui analizzati capiremo la connessione tra le dinamiche complesse non lineari e i nostri sistemi neurali. Ovviamente la struttura è basata su modelli nonlineari (solo in questo modo possiamo dare luogo alle dinamiche complesse) e sicuramente connessi localmente. La singola cellula non è troppo complicata da studiare: solo nel caso in cui viene connessa con la moltitudine otteniamo la dinamica complessa; ciò permette di generare le dinamiche auto organizzate, utili per l'azionamento di diversi tipi di attuatori. In quest'ottica vedremo i circuiti non lineari ed il relativo modelling → analisi della dinamica complessa in circuiti non lineari & localmente connessi.

L'idea finale sarebbe quella di ottenere un CPG, ovvero una rete neurale biologica che produce uscite ritmiche senza feedback sensoriale.

NB: il singolo neurone emette spike (è un oscillatore) ma quando è connesso con gli altri riesce a stimolare l'emersione di una complessiva dinamica ordinata che porta al controllo della locomozione.

Vediamo ora le equazioni che governano il sistema: Reaction Diffusion Equation (**equazioni differenziali alle derivate parziali**):

- $f(u)$ is the nonlinear part related to **time**, la quale include la parte dinamica che contiene la cinetica della reazione (reactive part). Here we can have the neuron equation (Izhichevic o Fitzhumo neuron)
- $D\nabla^2 u$ is the diffusion part and is responsible for the **space** dynamics (diffusion part). This part explains us how the neurons are connected together. La seconda formula ci mostra il "Laplacian Coupling": diffusione in spazio rispetto all'asse x e rispetto all'asse y.

$$\frac{\partial u}{\partial t} = f(u) + D\nabla^2 u$$
$$\nabla^2 u_i = \frac{\partial^2 u_i}{\partial x^2} + \frac{\partial^2 u_i}{\partial y^2}$$
$$u \in \mathbb{R}^m; f \in \mathbb{R}^m; D \in \mathbb{R}^{m \times m}$$

Reaction-diffusion Equations

Questa equazione è molto elegante poiché include in essa sia gli aspetti relativi al tempo, sia gli aspetti relativi allo spazio. Infatti questa equazione è detta reazione – diffusione poiché è suddivisibile in due termini: la prima è la cinetica di reazione. La seconda è la derivata parziale di “u” (“u” è intesa come variabile di stato generica, ad esempio il potenziale di tensione) e dunque la diffusione rispetto agli assi. Da quest'ultimo termine otteniamo la propagazione spaziale bidimensionale (derivata parziale rispetto ad “x” e “y”) e dunque la diffusione nel mezzo.

Dal momento che le celle realizzano una dinamica del secondo ordine, allora esse sono accoppiate (agganciate) solamente alle loro celle più vicine. Ne deriva che le connessioni tra neuroni sono sostanzialmente locali. Questa è un importante differenza con le NN convoluzionali.

Nelle equazioni precedenti si hanno i seguenti termini:

- “u” è un vettore m-dimensionale che definisce lo stato del sistema (dove x appartiene ad \mathbb{R}^m) nel tempo

- “ $f(u)$ ” è un set di m funzioni non lineari del vettore “ u ” delle variabili di stato e rappresenta un termine dinamico.
- D è una matrice $m \times m$. Gli elementi diagonali di D (ovvero D_i) sono conosciuti come i coefficienti di diffusione dell' i -esima variabile di stato

CELLULAR NEURAL NETWORKS

Vediamo ora le caratteristiche della rete neurale. Analizzeremo dunque un implementazione circuitale analogica delle CNN (non linear locally connected circuits).

L'unità circuitale base delle reti neurali è chiamata cella. Essa contiene elementi circuitali lineari e non lineari, i quali tipicamente sono capacitori lineari, resistori lineari, sorgenti controllate lineari e non lineari e sorgenti indipendenti.

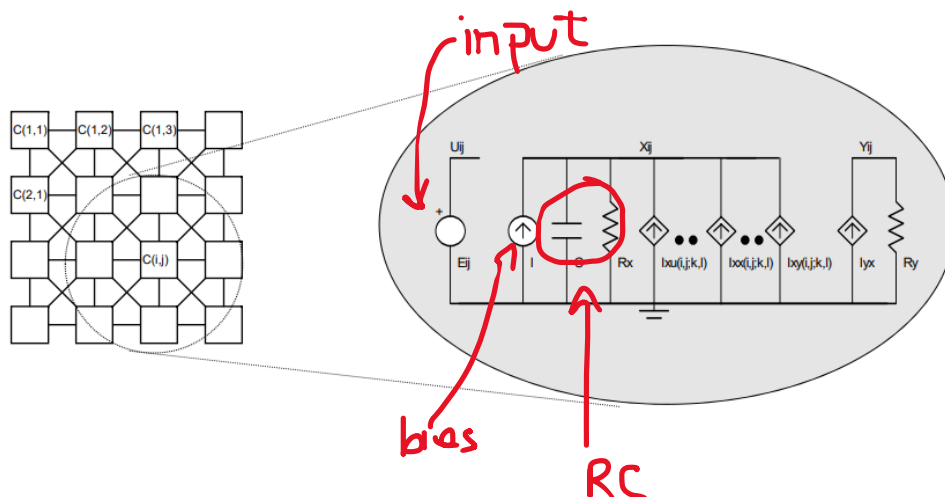
Le cellule adiacenti possono interagire direttamente con ciascun'altra. Le cellule non direttamente connesse insieme possono influenzare ciascun'altra indirettamente a causa degli effetti di propagazione della dinamica tempo-continua delle reti neurali cellulari.

Quando consideriamo una cellula, appartenente alla matrice di cellule, è come se prendessimo l'equazione Reaction Diffusion, la bloccassimo rispetto allo spazio (i,j) e considerassimo soltanto l'evoluzione nel tempo (per ogni quadratino della matrice).

Partiamo prima di tutto dall'elemento essenziale, ovvero il singolo neurone. Concentriamoci sulla singola cellula $C(i,j)$ in figura. La struttura base di $C(i,j)$ è un sistema dinamico che può essere rappresentato da circuiti nonlineari.

La struttura di base sarà strutturata come un circuito analogico : il vantaggio è poter andare molto veloci in analogico poiché non usiamo una sequenza di bit (il flow dei bit è il collo di bottiglia delle strutture neurali artificiali classiche). Leon Chua pensò che invece di usare un singolo microprocessore (ANN) fosse possibile utilizzare un array distribuito di semplici ma nonlinear processing unit (analogico). Il vantaggio è andare alla velocità che si desidera, in quanto “elettronica analogica” significa che l'evoluzione del tempo dipende dalle costanti di tempo.

NB: Ciò significa che se si vuole implementare una struttura del genere per processare un video, è possibile arrivare a quantitativi molto alti di fps.



Vediamo l'elettronica della generica cellula $C(i,j)$. La variabile di stato associata alla cellula singola è la tensione associata al capacitore del circuito (unica variabile di stato della cellula). La natura del circuito è RC.

Nella singola cellula, i suffissi u, x, e y denotano rispettivamente l'ingresso, lo stato e l'uscita.

I componenti essenziali del circuito sono la capacità (variabile di stato) e la resistenza R_x . Sostanzialmente la base del circuito è RC ma "arricchito" da altri componenti. Oltre a ciò, abbiamo:

- Tensione di nodo v_{xij} di $C(i,j)$ è chiamata lo stato della cella e la sua condizione iniziale deve essere tale da avere un'ampiezza minore o uguale a 1.
- un bias (generatore indipendente "I" di corrente in parallelo al circuito RC);
- Un generatore indipendente di tensione U_{ij} (disconnesso in figura) : rappresenta un input dall'esterno.
- Generatori di corrente controllati in tensione in parallelo. In questo caso, la corrente di questi generatori dipende dalla tensione delle cellule vicine:
 - $I_{xu}(l,j;k,l)$ è la nomenclatura utilizzata: "u" indica l'input delle cellule vicine, (i,j) è la cella considerata, (k,l) sono i pedici delle cellule vicine a quella considerata. La corrente di questo generatore pilotato associato alla nostra cella, dunque, dipende dagli input delle cellule vicine. Abbiamo dunque un numero di generatori pilotati pari al numero di cellule vicine alla cella che stiamo considerando (ricordiamo il concetto di connessione locale);
 - I_{xx} (generatori di corrente che dipendono dalla variabile di stato delle cellule vicine);
 - I_{xy} (generatori di tensione che dipendono dall'output dei neuroni vicini).
 - **Quindi il neurone (i,j) dipende dagli input, dalle variabili di stato e dagli output delle cellule vicine: la dipendenza viene esplicitata tramite i generatori pilotati (abbiamo dunque ben 3 tipologie di generatori di corrente pilotati).**
- La parte finale del circuito (I_{yx} e R_y) è la parte non lineare.

QUINDI: la seconda lettera nella struttura dei generatori pilotati del tipo " I_{x*} " rappresenta la fonte per pilotare il generatore → input delle cellule vicine, variabili di stato delle cellule vicine o output delle cellule vicine. Vedi in basso ulteriori specificazioni

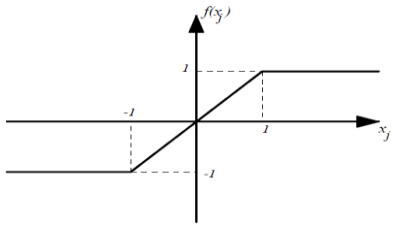
Nella figura accanto possiamo analizzare la natura e le dipendenze che generano le correnti dei generatori pilotati. Qui vengono specificati i pesi che moltiplicano i parametri delle cellule vicine ai fini di ottenere le correnti pilotate.

$$\begin{aligned} I_{xy}(i,j;k,l) &= A(i,j;k,l)y_{kl} \\ I_{xu}(i,j;k,l) &= B(i,j;k,l)u_{kl} \\ I_{xx}(i,j;k,l) &= C(i,j;k,l)x_{kl} \end{aligned}$$

cellule vicine

Anche in questa tipologia di rete neurale si ha il concetto di "weight"

(matrici A, B, C): essi rappresentano i pesi sinaptici che regolano la dinamica neurale. Proprio perchè le matrici esplicano le connessioni sinaptiche, allora parliamo di "**template values**" (A = feedback template, B = control template, C = state template). La corrente del generatore pilotato è dunque ottenuta dal prodotto dei pesi sinaptici per la variabile da considerare (variabile di stato, input o output delle **cellule VICINE alla nostra in analisi**). La corrente dei generatori pilotati è ottenuta dal prodotto dei pesi moltiplicati per le caratteristiche delle cellule vicine (input, output, vds). Se consideriamo lo schema circuitale della rete visto precedentemente, notiamo che abbiamo 9 (conteggio di tutte le cellule nell'architettura mostrata) diversi coefficienti per ogni tipologia di matrice (A,B,C). This means 27 voltage controlled current source and means 27 different synaptic connections from the neighbors to the cells i,j. This is the meaning of what we call Cellular nonlinear networks.



Esempio che descrive la non linearità (del tipo di saturazione). Questa non linearità è semplice da realizzare (e implementare) poiché si può considerare come la saturazione di un amplificatore operazionale (a causa del supply).

Vogliamo capire quale sia la relazione tra Y_{ij} (output) e X_{ij} (stato). La funzione "f", applicata allo stato X_{ij} , è non lineare.

LOCAL DYNAMIC OF THE SINGLE CELL

Possiamo dunque tentare di scrivere le equazioni e dunque la dinamica di ogni singola cellula.

$$C\dot{x}_{ij}(t) = -\frac{1}{R_x}x_{ij}(t) + \sum_{C(k,l) \in N_r(i,j)} A(i,j;k,l)y_{kl} + \sum_{C(k,l) \in N_r(i,j)} B(i,j;k,l)u_{kl} + \sum_{C(k,l) \in N_r(i,j)} C(i,j;k,l)x_{kl} + I$$

Handwritten notes in red:

- $\frac{1}{R_x}$ is labeled "RC".
- $x_{ij}(0) = x_{ij0}$ is labeled "initial cond.".
- $y_{ij} = \frac{1}{2} [|x_{ij} + 1| - |x_{ij} - 1|]$ is labeled "non linearity".
- The term $\sum C(i,j;k,l)x_{kl}$ is labeled "template".
- The bias term I is labeled "bias".

Possiamo dunque tentare di scrivere le equazioni di stato di ogni singola cellula.

Stiamo scrivendo le equazioni in corrente (infatti il risultato è dato dal prodotto del capacitore per la derivata della tensione).

L'equazione è data dalla somma della corrente sul resistore R_x , delle correnti relative ai generatori pilotati (dipendenza da neuroni adiacenti, moltiplicazioni per le matrici dei pesi sinaptici, template A B C) e infine del bias "I". **Dunque questa equazione è una sorta di legge di Kirchoff applicata al nodo.** $x_{ij}(0)$ rappresenta le condizioni iniziali. " y_{ij} " rappresenta la non linearità raffigurata precedentemente (saturazione).

Questa equazione rappresenta il comportamento del singolo neurone della rete neurale (tramite l'analisi del circuito proposto prima). Il vantaggio ricordiamo è l'implementazione su Analog Real Time Processors: upgrade in velocità.

NB: non vi sono algoritmi di learning per apprendere A, B, C. Tuttavia vi sono noti dei template possibili da utilizzare per image processing, signal processing, video filtering.

ES: ogni cellula è connessa ad un fotorecettore (variabile di stato connessa alla sensibilità della luce); abbiamo creato una CNN che è in grado di processare immagini (visual machine). Sensing e processing sono direttamente connessi e non provocano delay; subito dopo l'acquisizione, la singola cellula processa in real time grazie all'implementazione analogica.

NB: quando parliamo di "linear time invariant templates" significa che i parametri A,B,C non dipendono dal tempo e sono costanti.

IMPORTANTE: lezione 23/11/2020, 1h.20min, analisi accurata del circuito su whiteboard. riproposto anche in data 27/11/20 fino a min 20

AUTONOMOUS WAVE

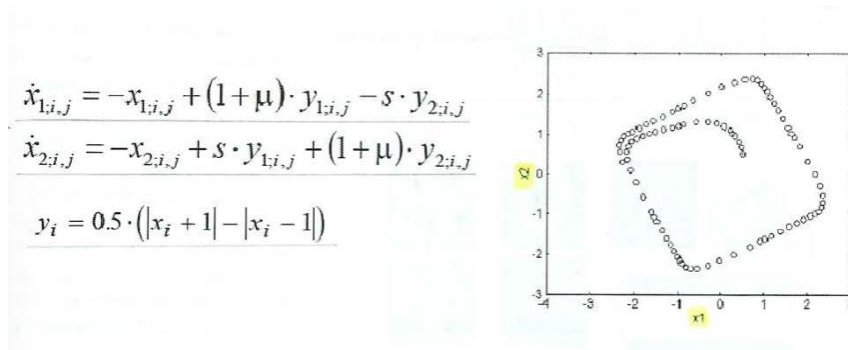
Onde che non richiedono nessun tipo di input, quindi autonome per definizione. Vediamone ulteriori caratteristiche:

- Ad esempio la propagazione degli spike negli assoni dipende dall'apertura dei canali ionici: è il medium che fornisce energia alla propagazione dell'onda.
- Nel caso di Autonomous Wave l'energia viene **estratta dall'ambiente** circostante tipicamente; dunque, la propagazione della wave non provoca perdita di energia. ES: il fuoco che si propaga e prende energia dall'ambiente circostante.
- Inoltre per le AW: ampiezza, shape e velocità sono costanti poiché dipendono dalla geometria del medium.
- Dopo una collisione abbiamo un annullamento delle onde ma non riflessione.
- La dinamica è slow fast.

Dopo questa introduzione, cerchiamo di capire come rappresentare questi fenomeni nei sistemi non lineari.

Stiamo dunque arrivando sino alla analisi delle onde autonome dentro reti neurali. Le CNN possono essere approssimate a sistemi del primo ordine per analisi semplici. Un esempio di sistema del primo ordine (generante spike) è quello di Izhikevich: tuttavia esso è molto povero a livello dinamico per generare dinamiche neurali.

Necessitiamo di un modello del secondo ordine.



A destra è possibile vedere le equazioni di una cellula della CNN con relativa non linearità (y).

Nel phase plane otteniamo la dinamica del sistema del secondo ordine come limit cycle: questa oscillazione è la traiettoria.

NB: la struttura è sostanzialmente regolare, non vi è molta differenza tra slow e fast dynamic.

L'attività di base ciclica di un oscillatore è pronta per spigare la dinamica neurale. Il comportamento ciclico dell'oscillatore esplica il neurone.

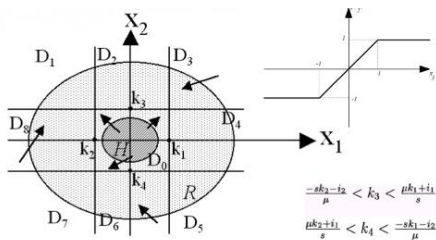
EXISTENCE of a PERIODIC ORBIT: Pointcarè – Bendixson Theorem

Questo teorema è un tool per stabilire l'esistenza di orbite periodiche in sistemi bidimensionali (ovvero la nostra CNN con due variabili di stato). Questo teorema, infatti, ci dà una **condizione sufficiente** per la generazione di un ciclo limite stabile.

Questo studio è importante per capire come le cellule generano un limit cycle.

Il teorema afferma: se esiste una regione R chiusa (come una circonferenza), la quale non contiene punti fissi (per il vector field $x=f(x)$), e se esiste all'interno di R una traiettoria C, allora R contiene un'orbita chiusa.

Dunque vogliamo trovare una regione chiusa nel piano delle fasi che non abbia punti di equilibrio interni e che abbia una traiettoria interna: in questo caso la traiettoria è un'orbita chiusa.



NB2: nella figura mostrata, l'anello interno serve ad escludere l'origine, il quale è considerato un punto fisso (per fare un esempio).

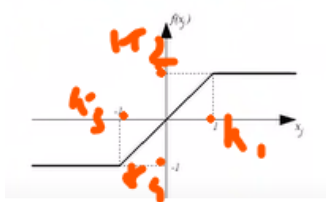
Per dimostrare che vi è un'orbita chiusa vediamo che tutte le traiettorie (indipendentemente da dove iniziano) sono intrappolate dentro R.

Per dimostrare che vi è un limit cycle, vogliamo ottenere che tutte le condizioni iniziali (**interne o esterne** che siano rispetto a R) vengano a confluire nella nostra traiettoria interna ad R (il limit cycle stesso). Inoltre tutto il flow è interno e non esce.

NB: nel phase plane in figura abbiamo le due variabili di stato X_1 e X_2 .

Le frecce in grassetto in figura mostrano che dal cerchio interno si va verso l'anello esterno grigio mentre dall'esterno della figura si va verso l'anello grigio: questa è la convergenza delle traiettorie di cui necessitavamo verso il limit cycle. Sia che ci troviamo esternamente o internamente all'anello, la traiettoria tende ad andare verso di esso. Il teorema che stiamo studiando aggiunge che, qualora non ci siano punti di equilibrio nell'anello, allora vi è un limit cycle.

Qualora il teorema sia verificato, possono accadere due opzioni equivalenti: o la traiettoria è essa stessa il limit cycle OPPURE equivalentemente la traiettoria va a spirale verso il limit cycle.



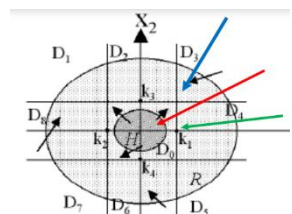
Nella figura vi sono dei parametri k_1, k_2, k_3, k_4 : questi sono i bound della linearità (oltre essi si ha saturazione).

Abbiamo due valori di K per ogni variabile di stato, per un totale di quattro.

NB: per implementare la non linearità semplice, deve accadere che $K_n=1$.

Dunque i valori di K servono solo a individuare la **"inner square region"**, dove tutto è lineare.

QUINDI la regione **centrale è lineare** (freccia rossa). Ovviamente bisogna considerare che può accadere che anche solo una variabile possa essere saturata (ES: se solo x_1 è maggiore di k_1 , mentre x_2 è interno al bound lineare). In questo caso si dice che il sistema è **parzialmente saturato**.



EQUAZIONI DEL MODELLO LIMIT CYCLE

The system:

$$\begin{aligned}\dot{x}_{1;i,j} &= -x_{1;i,j} + (1 + \mu) \cdot y_{1;i,j} - s \cdot y_{2;i,j} + i_1; \\ \dot{x}_{2;i,j} &= -x_{2;i,j} + s \cdot y_{1;i,j} + (1 + \mu) \cdot y_{2;i,j} + i_2; \\ y_i &= 0.5 \cdot (|x_i + 1| - |x_i - 1|)\end{aligned}$$

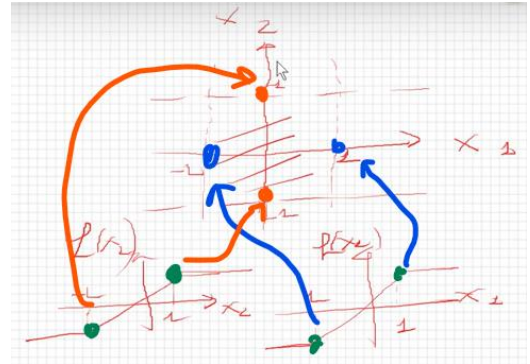
$(0 < \mu < s)$; \longrightarrow Stable limit cycle L

Nelle equazioni delle due variabili di stato si hanno i parametri "s" e "μ". Si ottiene che il limit cycle è **stabile** se "μ" è compreso tra zero e "s" ($0 < \mu < s$).

Analizziamo teoricamente il sistema.

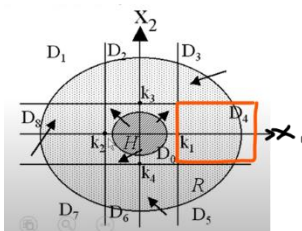
Siano date le nonlinearità delle funzioni delle variabili di stato x_1 e x_2 . Si esplicitino i punti delle non linearità nel phase plane (dati gli estremi della zona lineare nelle funzioni $f(x_1)$ e $f(x_2)$), li riportiamo nel phase plane), come mostrato in figura. Dunque nel phase plane individuiamo i punti $x = \pm 1$ e $y = \pm 1$.

Sostanzialmente stiamo vedendo un sistema uguale al precedente ma non lo stiamo visualizzando con i cerchi ma con queste quattro porzioni che interpolano i punti delle non linearità $f(x_1)$ e $f(x_2)$.



Per soddisfare il teorema da dimostrare, è necessario che non vi siano punti di equilibrio. Per fare ciò, dobbiamo dimostrare che i punti di equilibrio non sono reali ma virtuali.

// il concetto di punto di equilibrio virtuale: le traiettorie vanno verso il punto di equilibrio virtuale ma non ci arrivano (non hai capito altro). Sostanzialmente sono punti attrattori ma ai quali non si può arrivare. La spiegazione si ha nelle sbob anno 20-21. Tuttavia il concetto importante è quello che i punti di equilibrio virtuali sono punti attrattori che non posso raggiungere.



ESEMPIO: la regione segnata risulta essere saturata rispetto a X_1 e lineare rispetto X_2 .

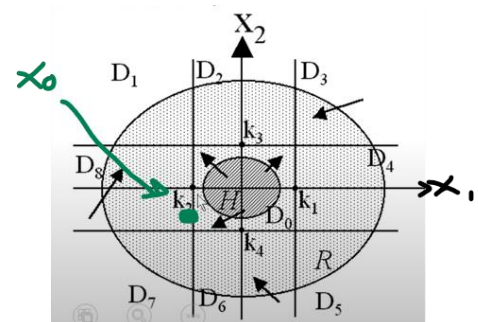
ESEMPIO DUE: le zone completamente non lineari sono D_1, D_3, D_7, D_6 .

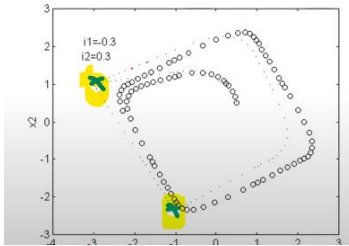
DIMOSTRAZIONE DELLA PRESENZA DI UNO STABLE LIMIT CYCLE L

Ipotesi e Tesi (vedi figura): ammettiamo che X_0 sia al confine tra due sottospazi lineari adiacenti, per uno dei quali X_0 è stabile; ammettiamo che X_0 appartenga a L (stable limit cycle). Allora il sistema mostra una dinamica slow fast.

Dimostriamo che è uno stable limit cycle con dinamica slow fast.

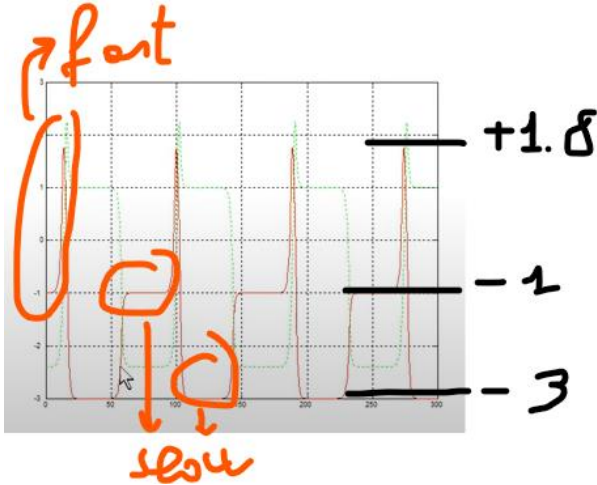
Prima di tutto aggiungiamo due termini di bias nelle equazioni di stato del modello (i_1 & i_2).





Simuliamo (o integriamo le variabili) con step costante: otterremo una distribuzione di punti. Questa distribuzione non è uniforme: come è possibile vedere dal plot, abbiamo degli addensamenti di punti che indicano un'accelerazione (considerando che appunto il time step è costante).

Applicando un bias ($i_1=0.3, i_2=0.3$), il comportamento qualitativo resta simile ma nei seguenti punti la traiettoria rallenta molto.



La traiettoria nel dominio del tempo è la seguente.

Vediamo le due variabili di stato e le relative dinamiche slow fast. Da -1 a +1.8 e poi indietro da +1.8 a -3 abbiamo una fast dynamic. Da -1 a -3 abbiamo la slow dynamic (nel tempo si presenta come un tratto piatto prima del prossimo spike da -3 a +1, poi piatto per un po' e poi un ulteriore picco verso +1.8).

Quindi essendo che il bias ha generato quei rallentamenti, allora nel tratto tra quei punti abbiamo una slow dynamic e dunque il bias ha complessivamente contribuito all'intera dicotomia slow fast.

DINAMICA SLOW FAST

Vediamo come ciò è possibile: dunque, dopo aver spiegato l'esistenza di un limit cycle, vediamo come si ottiene la dinamica slow fast:

Torniamo a dividere il piano in 9 parti in base alla linearità di X_1 e X_2 . Chiamiamo "Do" la regione lineare, "Ds" la regione di saturazione e "Dp" la regione di saturazione parziale.

Linearizziamo il sistema in queste 9 regioni per individuare i punti di equilibrio.

$$P_s^{++} : (1 + \mu - s + i_1, 1 + \mu + s + i_2); \quad P_s^{+-} : (1 + \mu + s + i_1, -1 - \mu + s + i_2)$$

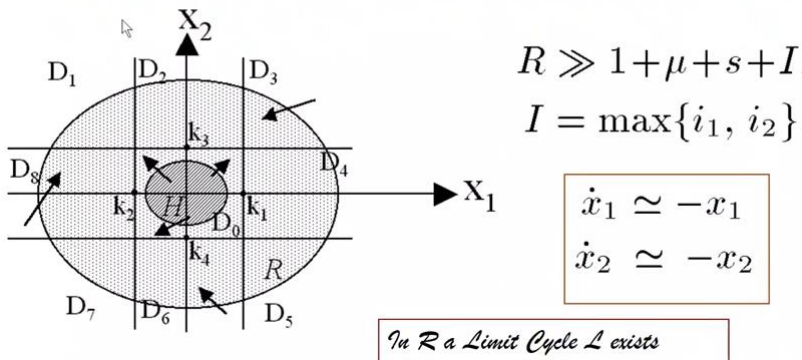
$$P_s^{--} : (-1 - \mu + s + i_1, -1 - \mu - s + i_2); \quad P_s^{-+} : (-1 - \mu - s + i_1, 1 + \mu + s + i_2)$$

Ogni punto di equilibrio risulta essere funzione di "mu" e "s". Variando i parametri dunque è possibile rendere i punti di equilibrio reali o virtuali.

DIMOSTRAZIONE LIMIT CYCLE

Il nostro obiettivo è individuare la regione del teorema di Poincaré dove dovrebbe trovarsi il ciclo limite. Quindi dobbiamo individuare le due circonferenze che delimitano l'anello.

$$J_0 = \begin{pmatrix} \mu & -s \\ s & \mu \end{pmatrix} \quad \begin{array}{l} \mathcal{H}: (0 < \mu < s) \\ \text{Unstable focus in the linear region} \end{array}$$



Occupiamoci di linearizzare il sistema nella parte lineare (J_0 è il jacobiano di D_0). Otteniamo che, qualora “mu” sia compreso tra zero e “s”, allora l’origine è un focus instabile (in generale ci riferiamo al cerchio grigio interno). Tutte le traiettorie vanno nell’anello esterno. Abbiamo trovato la circonferenza interna dell’anello. Inoltre a causa del punto instabile nel cerchio dell’ origine, allora le traiettorie vanno verso l’anello.

Per dimostrare il teorema di Pointcaré, è necessario individuare una circonferenza esterna di raggio R . Trovando l’anello, abbiamo trovato una condizione per il teorema di Pointcaré.

Se $R \gg 1 + \mu + s + I$, allora posso applicare x_1 e x_2 molto maggiori del valore $1 + \mu + s + I$. Otteniamo che y_2 e y_1 delle due equazioni di stato sono in saturazione. Significa che x_1 e x_2 sono maggiori rispetto all’altra parte dell’equazione: approssimiamo le variabili derivate ad esse stesse non derivate ($\dot{x}_1 \simeq -x_1$). Ciò significa che se siamo totalmente fuori dall’anello, sostanzialmente convergiamo verso il centro. Abbiamo trovato il secondo bound.

// qualche dubbio

Sostanzialmente abbiamo ottenuto che sia che siamo nel cerchio centrale, sia che ci troviamo esternamente a R , tutte le traiettorie convergono all’anello. Dunque all’anello, sicuramente si ha una traiettoria e le ulteriori traiettorie vi convergono.

La soluzione del sistema è il limit cycle stabile trovato precedentemente.

Ora si deve dimostrare che **il limit cycle è slow fast**. Sono due caratteristiche diverse ma che qui emergono contemporaneamente.

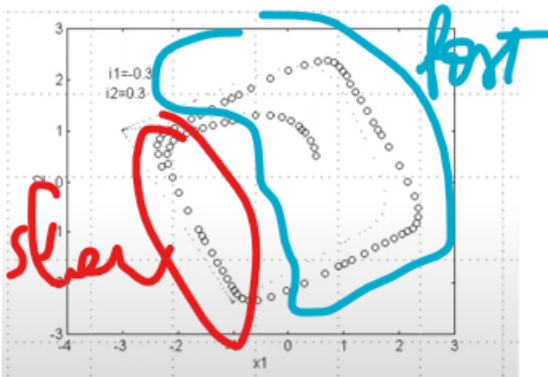
//spiega le motivazioni della slow fast dynamic, non capito

//lezione 12/04/2020: RECAP

RECAP

1. Modello di CNN 2-layer: un solo neurone dotato di un sistema non lineare del secondo ordine (due variabili di stato).
2. Si può dimostrare che sotto certe condizioni (Teorema), si ottiene un limit cycle di tipo slow fast.
3. Dividiamo il phase plane in 9 sub space: possiamo calcolare i punti di equilibrio.
4. Utilizzando il concetto di punti di equilibrio virtuali, tentiamo di spostarli sempre tra i sub space in modo tale che la nostra traiettoria li “seguirà” e genererà il limit cycle.

- Quando il punto virtuale si trova al confine e contestualmente la traiettoria sta arrivando su di esso, la traiettoria rallenta. Ciò accade poiché quando la traiettoria si sposta verso il punto di equilibrio, prima segue una dinamica veloce e man mano che si avvicina poi rallenta.
- Dopo di che si ottiene un jump che genera la caratteristica dello spike. Il jump avviene poiché rispetto ad una variabile sono in equilibrio ma non rispetto all'altra (caso in cui la traiettoria si trova al boundary).
- Abbiamo creato lo spiking neurale
- Lo spiking è controllato dai parametri, poiché i punti di equilibrio dipendono da essi (in particolare i_1 , i_2). Modulando dunque questi parametri si modula il ratio tra la phase slow e fast. In questa maniera si modula la spiking activity.



//circuit implementation

//reaction diffusion equations

La parte reattiva rappresenta la dinamica della cellula mentre la parte diffusiva è rappresentata dal laplacian coupling.

Vogliamo unire i neuroni tramite una sinapsi elettrica bidirezionale (come nell'esempio del clone).

Le equazioni diventano pari alla dinamica della cellula stessa sommata al "laplacian template" che rappresenta la diffusione dell'output. Dunque per ogni equazione abbiamo una parte di reaction e una parte di diffusion.

Ogni layer diffonde indipendentemente.

$$\begin{aligned}\dot{x}_{1,i,j} &= -x_{1,i,j} + (1 + \mu)y_{1,i,j} - sy_{2,i,j} + i_1 + D_1 \cdot (y_{1,i+1,j} + y_{1,i-1,j} + y_{1,i,j-1} + y_{1,i,j+1} - 4y_{1,i,j}); \\ \dot{x}_{2,i,j} &= -x_{2,i,j} + sy_{1,i,j} + (1 + \mu)y_{2,i,j} + i_2 + D_2 \cdot (y_{2,i+1,j} + y_{2,i-1,j} + y_{2,i,j-1} + y_{2,i,j+1} - 4y_{2,i,j});\end{aligned}$$

Il fattore D_1, D_2 rappresenta il fattore di diffusion coupling: parametro che funziona da peso per la funzione di diffusione rispetto alla funzione di reazione. Si usano piccoli valori di D (0.1), in modo tale da ridurre la

diffusione rispetto alla reazione: vogliamo diffondere poca energia generata nella reaction part (questa energia verrebbe facilmente persa).