Fourier Neural Operator for learning the dynamic of Ionic models

Luca Pellegrini, Massimiliano Ghiotto

December 23, 2024

Todo list

1 Automated hyper-parameters tuning

We have created the dataset as described in Section 2.1 and then we have split it into three parts with the rule 80/10/10, in particular:

- Training set (80% of the total dataset): we use this set to train our model.
- Validation set (10% of the total dataset): we use this set to compare different hyperparameter configurations and choose the best.
- Test set (10% of the total dataset): the model with the lower validation error is finally trained on this benchmark and the test errors are reported in this paper.

All the datasets are batched with mini-batches of 32 examples each.

We have automated hyper-parameter tuning using the Ray library to ensure that our neural operator model is as good as possible. We optimize the hyperparameters with HyperOptSearch, which uses the Tree-structured Parzen Estimators algorithm for 200 trials. This optimization process is completely distributed and parallel and can find an optimal configuration for a given model.

The training of the models is performed with ADAMW optimizer, with a learning rate equal to η and weight decay regularizer ω for a total of 1000 epochs and minimizing the L^2 relative error. We also use a step learning rate scheduler, reducing η of a factor to γ every $n_{step} = 10$ epochs. We take η and γ as hyper-parameters of the training process that have to be tuned. The objective of hyper-parameters tuning of the Fourier Neural Operator is the hidden dimension (d_v), the number of hidden layers (L), the number of Fourier modes considered (k_{max}), the activation function (σ), the number of padding points (n_{pad}) and the Fourier architecture modification. The best configurations of the hyper-parameters for each benchmark are listed in Table 1.

problem	η	ω	γ	d_v	L	k_{max}	σ	n_{pad}	FNO_mod
FHN	$7.1e^{-4}$	$3.4e^{-4}$	0.91	256	5	8	leaky_relu	0	Residual
HH	$7.5e^{-4}$	$3e^{-6}$	0.92	256	5	8	leaky_relu	7	Zongyi

Table 1: Your caption here

2 FitzHugh-Nagumo

$$\begin{cases} \dfrac{dV}{dt} = bV(V-\beta)(\delta - V) - cw + I_{app}, & t \in [0,T] \\ \dfrac{dw}{dt} = e(V - \gamma w), & t \in [0,T] \\ V(0) = V_0, w(0) = w_0 \end{cases}$$

Where

- b = 5
- β = 0.1
- c = 1
- δ = 1
- γ = 0.25
- e =1

Our objective is to learn the operator

LP: Regolarita' di HH,FHN

$$\begin{aligned} \mathcal{G}^\dagger : &\mathbb{R} \to H^1([0,T];\mathbb{R}) \times H^1([0,T];\mathbb{R}) \\ &I_{app}(t) \mapsto (V(t), w(t)) \end{aligned}$$

2.1 Dataset

The data set is created using Matlab ode15s with $T = 1$. We randomly choose the intensity of the current and the duration of the stimulus T_{stim}. In order to get all the important dynamics of the system, the generation of the dataset for training is divided into three parts formed in according to the following table:

Name	Range of values for the current	Range of values for T_{stim}	$n_{examples}$
t_0	$(0.1, 2)$	$(0, 0)$	20
General	$(0.1, 2)$	$(0.01, 1)$	2300
nap	$(1e-4, 0.01)$	$(0.1, 1)$	500

For testing, we generate 375 examples where the dataset is split into four.

Name	Range of values for the current	Range of values for T_{stim}	$n_{examples}$
t_0	$(0.1, 2)$	$(0, 0)$	10
General	$(0.1, 2)$	$(0.01, 1)$	285
nap	$(1e-4, 0.01)$	$(0.1, 1)$	30
t_{fin}	$(0.3, 2)$	$(1, 1)$	50

Here we report the values of the loss function over the epochs

In figure 2 we present the results obtained by taking five randomly selected examples for each part of the dataset (General, nap, and t_{fin}). (non messi i t_0 perchè è semplicemente la soluzione nulla, qui abbiamo riportato gli esempi con gli errori ma si possono avere li stessi esempi con invece che l'errore lo spazio delle fasi)

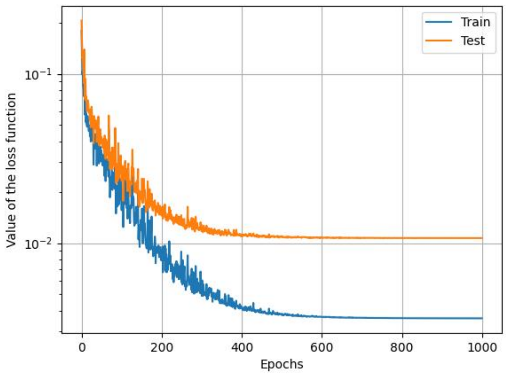
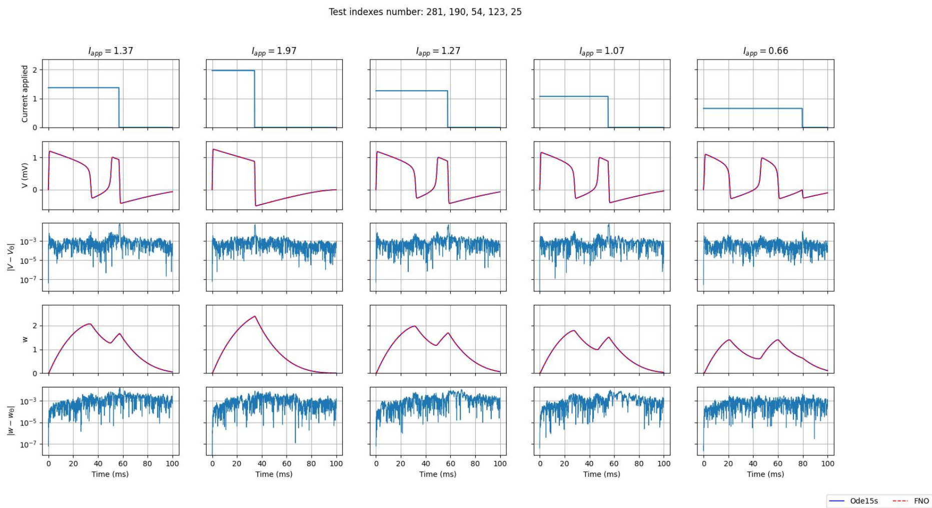
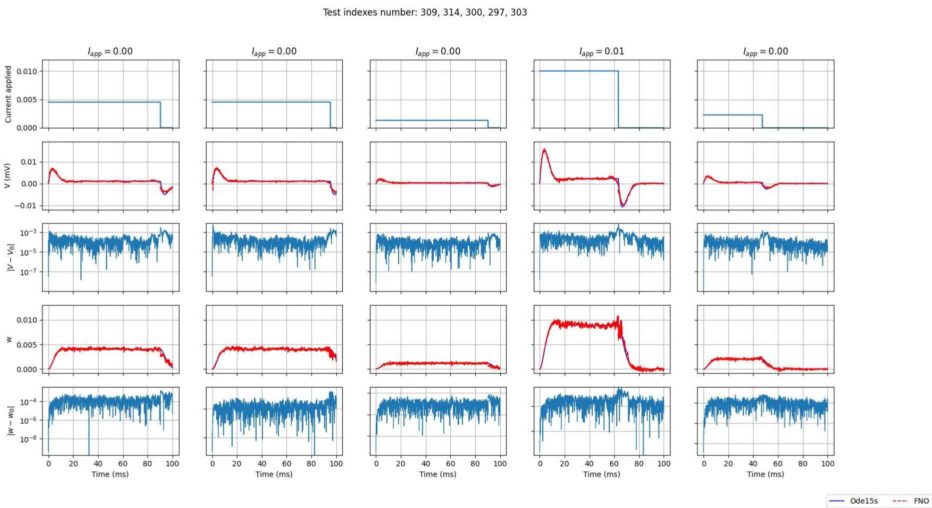


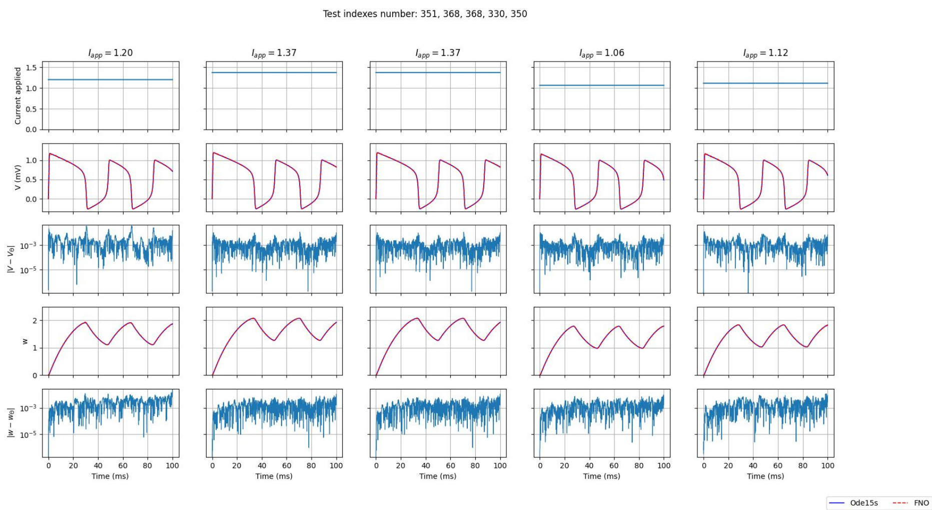
Figure 1: Value of the loss function over the epochs



(a) Five examples for the general part of the dataset



(b) Five examples for the nap part of the dataset



(c) Five examples for the t_{fin} part of the dataset

Figure 2: Each figure contains five plots, arranged in order from the top row as follows: current applied, voltage, point-wise error for the voltage, recovery variable w, and point-wise error for the recovery variable.

3 Hodgkin-Huxley

$$
\begin{cases}
C_m \dfrac{dV}{dt} = -\Big(\bar{g}_{Na} m^3 h (V - V_{Na}) + \bar{g}_K n^4 (V - V_K) + \bar{g}_L (V - V_L)\Big) + I_{app}, & t \in [0, T], \\[2ex]
\dfrac{dm}{dt} = \alpha_m(V)(1 - m) - \beta_m(V) m, & t \in [0, T], \\[2ex]
\dfrac{dh}{dt} = \alpha_h(V)(1 - h) - \beta_h(V) h, & t \in [0, T], \\[2ex]
\dfrac{dn}{dt} = \alpha_n(V)(1 - n) - \beta_n(V) n, & t \in [0, T], \\[2ex]
V(0) = 2.7570e - 02, \\[2ex]
m(0) = 5.2934e - 02, \\[2ex]
h(0) = 5.9611e - 01, \\[2ex]
n(0) = 3.1768e - 01.
\end{cases}
$$

Where

- $C_m = 1$
- $\bar{g}_{Na} = 120$
- $\bar{g}_K = 36$
- $\bar{g}_L = 0.3$
- $V_{Na} = 115$
- $V_K = -12$
- $V_L = 10.6$

$$
\begin{aligned}
\alpha_m(V) &= 0.1(25 - V)\Big[\exp\Big(\frac{25 - V}{10}\Big)\Big]^{-1} & \beta_m(V) &= 4\exp\Big(-\frac{V}{18}\Big) \\
\alpha_h(V) &= 0.07\exp\Big(-\frac{V}{20}\Big) & \beta_h(V) &= \Big[\exp\Big(\frac{30 - V}{10}\Big)\Big]^{-1} \\
\alpha_n(V) &= 0.01(10 - V)\Big[\exp\Big(\frac{10 - V}{10}\Big) - 1\Big]^{-1} & \beta_n(V) &= 0.125\exp\Big(-\frac{V}{80}\Big)
\end{aligned}
$$

$$
\begin{aligned}
\mathcal{G}^\dagger : &\mathbb{R} \to H^1([0, T]; \mathbb{R}) \times H^1([0, T]; \mathbb{R}) \times H^1([0, T]; \mathbb{R}) \times H^1([0, T]; \mathbb{R}) \\
&I_{app}(t) \mapsto (V(t), m(t), h(t), n(t))
\end{aligned}
$$

3.1 Dataset

The dataset is formed using odes15s of Matlab with $T = 1$. Where we randomly pick the intensity of the current and the duration of the stimulus T_{stim}. In order to get all the important dynamics of the system the generation of the dataset for the training is split in three formed in the following way (i nomi sono quelli usati su matlab)

Name	Range of sampling for the current	Range of sampling for T_{stim}	$n_{examples}$
t_0	$(2, 10)$	$(0, 0)$	20
General	$(2, 10)$	$(0.01, 1)$	2380
nap	$(1e - 4, 2)$	$(0.1, 1)$	100
i_{high}	$(50, 200)$	$(0.1, 1)$	300
t_{fin}	$(2, 30)$	$(1, 1)$	200

For the testing we genera 375 example where the dataset is splitted in four.

Name	Range of sampling for the current	Range of sampling for T_{stim}	$n_{examples}$
t_0	$(0.1, 2)$	$(0, 0)$	10
General	$(0.1, 2)$	$(0.01, 1)$	275
nap	$(1e-4, 2)$	$(0.1, 1)$	30
i_{high}	$(50, 200)$	$(0.1, 1)$	30
t_{fin}	$(2, 30)$	$(1, 1)$	30

Here we report the values of the loss function over the epochs

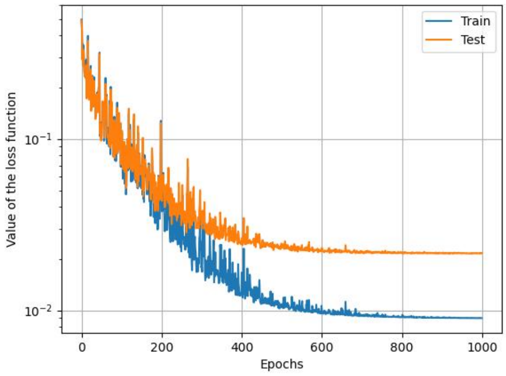
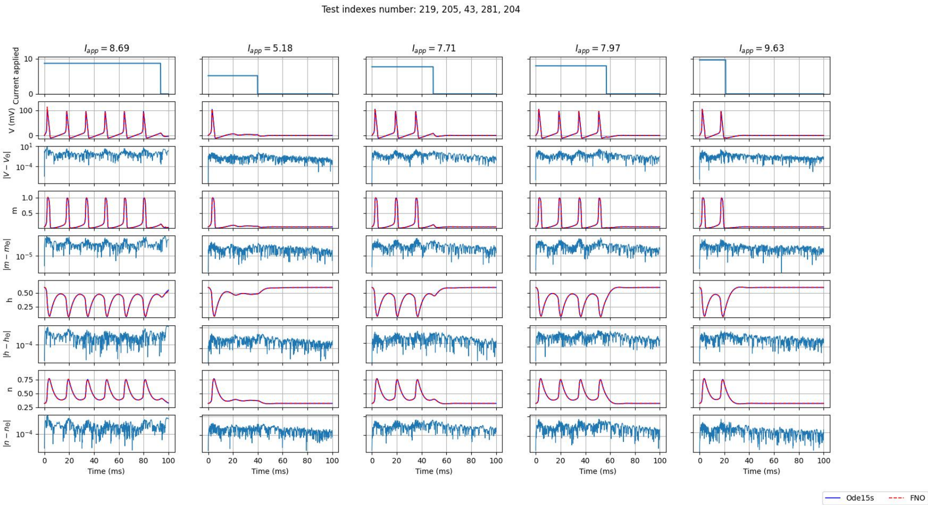
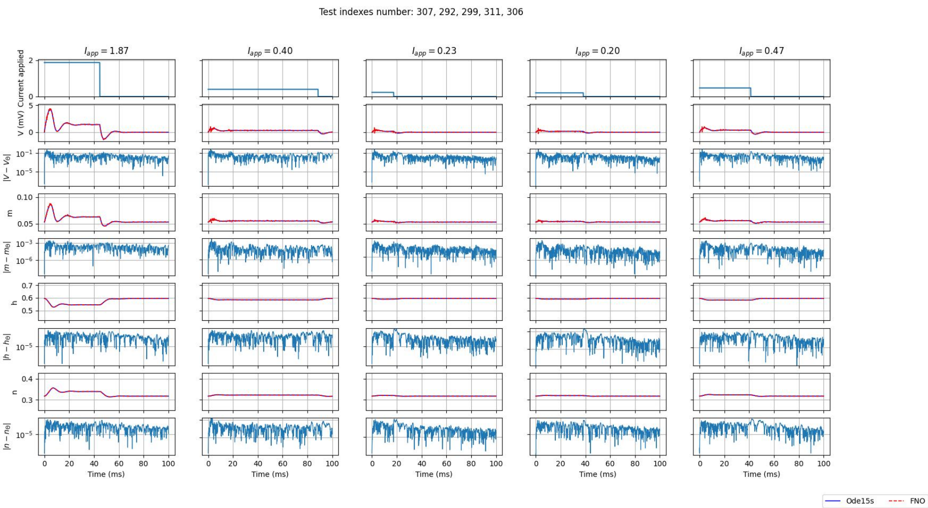


Figure 3: Value of the loss function over the epochs

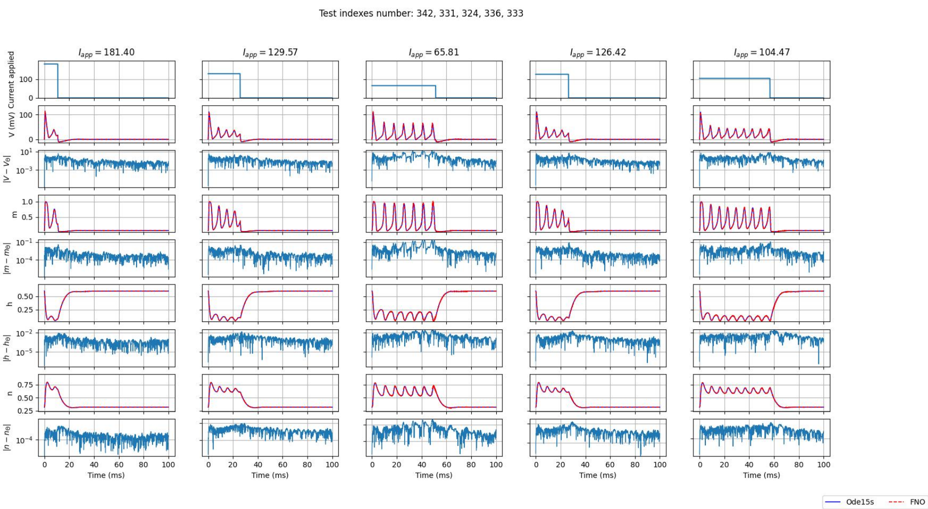
As before we take five examples for each part of the test dataset (General,i_{high},nap,t_fin) and visualize the results in figure 5.



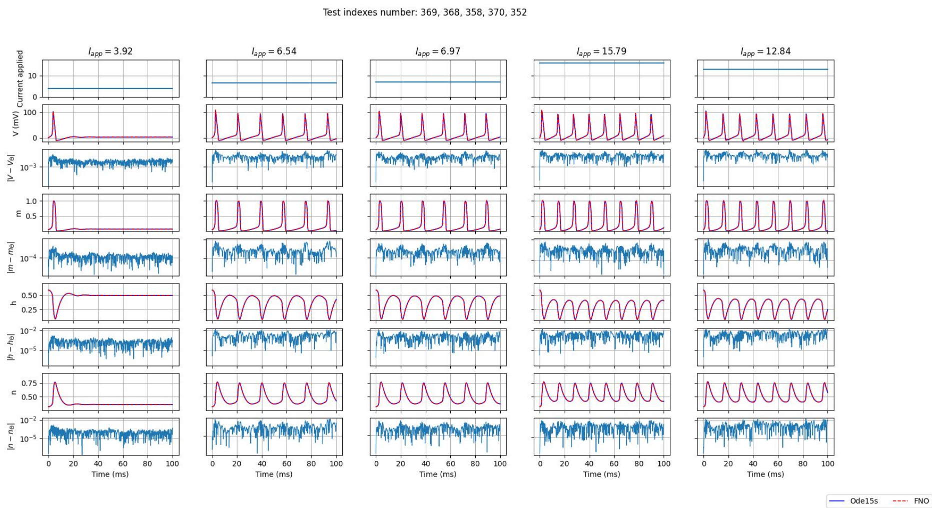
(a) Five examples for the general dataset



(b) Five examples for the nap dataset



(c) Five examples for the i_{high} dataset



(a) Five examples for the t_{fin} dataset

Figure 5: Results obtained each of the figures has the following graph in order from the top row: current applied, Voltage, point-wise error for the Voltage,gating variable m, point-wise error for m,gating variable n,point-wise error for n,gating variable h, error for the gating variable h

4 Luo Rudy

$$\begin{cases} \dfrac{dV}{dt} & = \bar{g}_{Na} m^3 hj(V - V_{Na}) + \bar{g}_{si} df(V - Vsi) + \bar{g} XX_i(V - V_K) + \bar{g}_{K1} K1_{\infty}(V - V_{K1}) + \\ & + \bar{g}_{Kp} K_p (V - V_{K_p}) + \bar{g}_b (V - V_b) \\ \dfrac{dm}{dt} & = \alpha_m (1 - m) + \beta_m m \end{cases}$$