By Boshuo Wang, March 16, 2018

Summary

This repository contains NEURON and MATLAB code and data for the article titled "Modified cable equation incorporating transverse polarization of neuronal membranes for accurate coupling of electric fields" (Wang *et al.*, 2018, *J. Neural Eng.*, 15 026003. Referred to as "the paper" in the following text). The code simulates various neural stimulation situations with uniform field or electrodes and investigates how much the electric field (E-field) component transverse to the neurons affects activation thresholds.

Model set up

The models are typically coded by strings in the format of `SOURCE_TARGET_MEMBRANE`, The number of parameters tested is determined by the specific combination of the set-up. The pulse duration for all sources are between 1 μs to 10 ms with 6 values each decade (a total of 25 pulse durations) except for the validation with linear membrane, which used pulse durations between 0.1 μs to 100 ms (a total of 37 pulse durations).

- `SOURCE` is the type of stimulation source.
  - **`UF`**: **`U`**niform **`F`**ield. Used with single compartments and axonal terminals and bends.
  - **`PE`** and **`DE`**: **`P`**oint **`E`**lectrode and **`D`**isk **`E`**lectrode. Used with long axons.
- `TARGET` is the type of neural element to be activated:
  - **`Axon`** and **`Soma`**: Single compartment of cylindrical (**`Axon`**, radius between 0.25 μm to 8 μm) or spherical (**`Soma`**, radius between 0.5 μm to 16 μm) geometry. Modeled with both NEURON and MATLAB. Total number of parameters: 150 = 6 × 25
  - **`TrmB`** and **`TrmS`**: Axonal **`Term`**inals with no synapse (**`B`**are, both HH and RMG) or with **`S`**ynapse (only for RMG). Activation is determined for antidromic propagation of action potential. Orientation of uniform field is varied between 0° and 90° with 24 angles and denser sampling near 90°. Total number of parameters: 600 = 24 × 25
  - **`BndO`** and **`BndA`**: Axonal **`Bend`**s, with activation determined for either **`O`**rthodromic and **`A`**ntidromic propagation. Terminals of axons are specifically deactivated so that threshold is only determined for mid-axon activation. Orientation of uniform field is varied between 0° and 135° with 24 angles and denser sampling between 100° and 120° (see Fig. 5 of paper for definition of orientation). Total number of parameters: 600 = 24 × 25
  - **`A`**: long **`A`**xons, with terminals specifically deactivated so that threshold is only determined for mid-axon activation. Axon–electrode distance varied between 4.5 μm and 10 cm and between 31 μm and 3.1 cm, with 6 values per decade for the linear membrane and nonlinear membrane models, respectively. Total number of parameters: 999 = 27 × 37 (linear) and 475 = 19 × 25)
- `MEMBRANE` is the membrane model used.

- **Lin**: unmyelinated axon with **Lin**ear membrane based on Hudgkin-Huxley model at rest (Hudgkin and Huxley, 1952). Used only with point electrode and disk electrode.
- **HH**: unmyelinated axon with **H**udgkin-**H**uxley ion channels adjusted to room temperature (**H**udgkin and **H**uxley, 1952). Used for all types of `SOURCE`s and `TARGET`s.
- **RMG**: myelinated axon of human peripheral nerve at body temperature (**R**ichardson, **M**cIntyre, and **G**rill, 2000). Used for all types of `SOURCE`s and `TARGET`s, except the single compartments.

The models can be separated into two classes (1st level folders): the first for validation of the modified cable equation (Section 3 of the paper) and the second for studying the effect of transverse polarization on activation threshold (Section 4 of the paper). A total of 5 types of simulation set-ups were used:

1. **UF_Axon_HH** and **UF_Soma_HH**: validation of the two implementation methods for modified cable equation using single compartments
2. **PE_A_Lin** and **DE_A_Lin**: validation of modified cable equation solver using distributed cable
3. **UF_TrmB_HH**, **UF_TrmB_RMG**, and **UF_TrmS_RMG**: axonal terminals and uniform field stimulation
4. **UF_BndO_HH**, **UF_BndA_HH**, **UF_BndO_RMG**, and **UF_BndA_RMG**: axonal bends and uniform field stimulation
5. **PE_A_HH**, **DE_A_HH**, **PE_A_RMG**, and **DE_A_RMG**: long axon and electrode stimulation

Units

The unit system is used in the simulations is given as follows. (Sometimes units are converted, but only for text display and in figures.)

• Temperature	degree Celsius	
• Length	centimeter	1 cm = 10 mm = 10^4 µm
• Time and frequency	millisecond and kilohertz	1 ms = 1 kΩ·µF = 1 kHz^{-1}
• Potential	millivolt	1 mV = 1 kΩ·µA = 1 ms·µA·$\mu\mathrm{F}^{-1}$
• Current	microampere	1 µA = 1 mS·mV
• Resistance and conductance	kiloohm and millisiemens	1 kΩ = 1 mV·$\mu\mathrm{A}^{-1}$ = 1 mS^{-1}
• Capacitance	microfarad	1 µF = 1 ms·µA·mV^{-1}
• Charge	nanocoulomb	1 nC = 1 ms·µA = 1 mV·µF

Math of cable equation solver

simulate_SC_HH.m, **simulate_cable_HH.m**, and **simulate_cable_RMG.m**

For cable with linear topology (i.e., no branching), the transversely-averaged membrane potential $\bar{\varphi}_\mathrm{m}$ along the cable (z direction) is discretized in space and time:

$$\bar{\varphi}_\mathrm{m}(z,t) \to \bar{\varphi}_\mathrm{m}(z_n,t_k) \to \bar{\varphi}_\mathrm{m}^k[n]$$

The discretization is not necessary uniform, i.e., compartment length Δz_n and time step Δt^k may vary. The cable has N compartments, each with radius R_n, surface area $A_n = 2\pi R_n \Delta z_n$, specific membrane capacitance c_n, and axial resistances to its previous/left (−) and next/right (+) neighbor $R_{\mathrm{i},-}[n]$ and $R_{\mathrm{i},+}[n]$

$$R_{\mathrm{i},\pm}[n] = \rho_{\mathrm{i}}\left(\frac{\Delta z_n}{2\pi R_n^2} + \frac{\Delta z_{n\pm1}}{2\pi R_{n\pm1}^2}\right)$$

The outward axial resistances ($R_{\mathrm{i},-}[1]$ and $R_{\mathrm{i},+}[N]$) would contain only one term, but can be set to 0, NaN, or $+\infty$ since they are irrelevant in the cable equation calculation.

The compartments have the following time- and/or location-dependent parameters at each time step:

- Equivalent channel conductance and reversal potential $\bar{g}_n^k$ and $\bar{\mathcal{E}}_n^k$
$$\bar{g}_n^k = \sum_{b,j} g_{b,n}^{(j),k}\, , \bar{\mathcal{E}}_n^k = \frac{1}{\bar{g}_n^k}\sum_{b,j} \mathcal{E}_{b,n}^{(j),k} \cdot g_{b,n}^{(j),k}$$
The equivalent channels takes into account the averaging of different ion channels (indexed by j), and, for the modified cable equation, the discretization of the membrane in the transverse dimension (indexed by b).
- Equivalent time "constant" $\bar{\tau}_n^k$
$$\bar{\tau}_n^k = c_n / \bar{g}_n^k$$
- Left and right length "constants" $\bar{\lambda}_-^k[n]$ and $\bar{\lambda}_+^k[n]$
$$\bar{\lambda}_\pm^k[n] = \left(A_n \bar{g}_n^k R_{\mathrm{i},\pm}[n]\right)^{-\frac{1}{2}}$$
The length constants $\bar{\lambda}_-^k[1]$ and $\bar{\lambda}_+^k[N]$ do not appear in the backward Euler calculation of the cable equation.

The backward Euler equation for the entire cable is given in matrix form

$$\left(\bar{\boldsymbol{\tau}}^{\boldsymbol{k}} + \Delta t^k\left(\boldsymbol{I}_N - \boldsymbol{\Lambda}^k\right)\right)\bar{\boldsymbol{\varphi}}_{\mathrm{m}}^{k+1} = \bar{\boldsymbol{\tau}}^{\boldsymbol{k}}\bar{\boldsymbol{\varphi}}_{\mathrm{m}}^{k} + \Delta t^k\left(\bar{\boldsymbol{\mathcal{E}}}^k + \boldsymbol{\Lambda}^k \bar{\boldsymbol{\varphi}}_{\mathrm{e}}^{k}\right)$$

The column vectors are $\bar{\boldsymbol{\varphi}}_{\mathrm{m,e}}^k = \left[\bar{\varphi}_{\mathrm{m,e}}^k[n]\right]_{N\times1}$ and $\bar{\boldsymbol{\mathcal{E}}}^k = [\bar{\mathcal{E}}_n^k]_{N\times1}$, the diagonal matrices are $\bar{\boldsymbol{\tau}}^{\boldsymbol{k}} = \mathrm{diag}\left(\bar{\tau}_n^k\right)$ and $\boldsymbol{I}_N = \mathrm{diag}(1,1,\dots,1)$, and the tridiagonal matrix $\boldsymbol{\Lambda}^k$ is given by

$$\boldsymbol{\Lambda}^k = \left[\Lambda_{i,j}^k\right]_{N\times N^-} = \begin{cases} \left(\bar{\lambda}_-^k[n]\right)^2 & i = j+1 = n \\ \left(\bar{\lambda}_+^k[n]\right)^2 & i = j-1 = n \\ -\left(\bar{\lambda}_-^k[n]\right)^2 - \left(\bar{\lambda}_+^k[n]\right)^2 & i = j = n \neq 1, N \\ -\left(\bar{\lambda}_+^k[n]\right)^2 & i = j = n = 1 \\ -\left(\bar{\lambda}_-^k[n]\right)^2 & i = j = n = N \end{cases}$$

The $\bar{\boldsymbol{\mathcal{E}}}^k$ term represents the ionic currents, the $\bar{\boldsymbol{\varphi}}_{\mathrm{e}}^k$ term represents the activating function of the extracellular field, and the remaining terms containing $\bar{\boldsymbol{\varphi}}_{\mathrm{m}}^k$ and $\bar{\boldsymbol{\varphi}}_{\mathrm{m}}^{k+1}$ represent the repolarization and propagation of membrane potentials. The first and last terms of the main diagonal of $\boldsymbol{\Lambda}^k$ are different to other terms, which reflect sealed boundary condition at the cable's terminals.

The matrix $\bar{\boldsymbol{\tau}}^{\boldsymbol{k}} + \Delta t^k\left(\boldsymbol{I}_N - \boldsymbol{\Lambda}^k\right)$ on the left-hand side is tridiagonal and $\bar{\boldsymbol{\varphi}}_{\mathrm{m}}^{k+1}$ can be solved using the tridiagonal algorithm with $O(N)$ run-time complexity, which can also handle the special case of single compartment (**SC**) in which all matrices and vectors become scalars. If branching is included, the $\boldsymbol{\Lambda}^k$ matrix contains off-diagonal terms for branch compartments, but the total number of non-zero terms is unchanged due to the tree-like topology. The backward Euler step would require a more general Gaussian elimination algorithm but does not need to calculate the inverse of the left-hand side matrix, and therefore run times still scales linearly with the number of compartments.

File organization

Folders and files are marked with black bullets ● and white bullets ○, respectively.

- **1 - Validation of modified cable equation**: contains code and data for the validation, (except the single compartment models in MATLAB, which are grouped with all other nonlinear membrane models in MATLAB for convenience)
  - **1 - Single compartment – NEURON**: NEURON code and data for single compartment validation, and MATLAB code for processing the results.
    - **main_UF_Axon_HH.hoc**: runs all simulation for axon
    - **main_UF_Soma_HH.hoc**: runs all simulation for axon
    - **global_parameter.hoc**: creates global parameter for the simulation
    - **create_cell.hoc**: specifies topological and biophysical properties of the cell, geometry of each compartment is default
    - **set_morph_func.hoc**: procedure to change morphology of the cell for IP
    - **set_time_vec.hoc**: procedure set time vector and stimulus vector
    - **stimulate.hoc** : function: runs simulation, determines whether cell fires
    - **find_thresh.hoc function**: threshold search, calls stimulate
    - **parameter_loop.hoc**: Main parameter loop to search threshold for all pulse durations
    - **hhh.mod**and related files: modified the original hh.mod file to include a large range of membrane potentials for the look-up table.
    - **process_NEURON_data.m**: processes all NEURON data and generates figures, including **Fig. 2** of the paper.
    - **NEURON Results**: folder containing all the results generated by **main_UF_Axon_HH.hoc** and **main_UF_Soma_HH.hoc**. Includes two subfolders.
    - **MATLAB_functions**: folder containing addition MATLAB functions used by **process_NEURON_data.m**
    - **Processed data and figures**: folder containing MATLAB data and figures generated by **process_NEURON_data.m**
  - **2 - Distributed linear cable - MATLAB**: code and data for validating the decoupling of the longitudinal and transverse dimension as well as the cable equation solver for distributed cable.
    - **main_linear_membrane.m**: main function. Runs simulation and generates figures, including **Fig. 4** of the paper.
    - **ES_A_Lin.m**: simulation with either electrode stimulation

    - **`tridiag.m`**: tridiagonal algorithm
    - **`plot_Schnabel2001.m`**: plots results in format of Fig.2 of Schnabel and Struijk, 2001
    - **`plot_Neu2016.m`**: plots results in format of Fig. 3 of Neu, 2016
    - **`figure_format.m`**: specifies formats for figures
  - **`Results`**: Results of simulation
  - **`Figures`**: Generated figures, including comparison with original figures from Schnabel and Struijk 2001 and Neu 2016.
- **`2 - Effect of transverse polarization on neural activation threshold`**: all nonlinear membrane models in MATLAB
  - **`main_SC_HH.m`**: main function to obtain thresholds for single compartment models with modified cable equation. . `results = main_SC_HH(mod_prmtr, out_ctrl)` Call function in a loop (parallelized) or run on computer cluster to cover all parameters `id` (see below). Input arguments:
    - `mod_prmtr`: structure specifying model parameter. Fields:
      - `model_name`, a string `'UF_Axon_HH'` and `'UF_Soma_HH'`.
      - `id`: integer, parameter ID. Range: 1-150.
    - `out_ctrl`: structure specifying outputs. Fields of logical or 0/1:
      - `if_save_data`: whether to save the output `results` in a .mat file in a subfolder for each simulation.
      - `if_write_log`: whether to write threshold finding process in a .txt log in a subfolder for examining search process.
      - `if_plot`: whether to plot threshold finding process in a subfolder for visual confirmation of threshold values.

    Output arguments:

    - `results`: structure containing the simulation results. Fields:
      - `th_MCE`: threshold obtained with the modified cable equation.
  - **`main_ES_HH.m`** and **`main_ES_RMG.m`**: main functions to obtain thresholds for all distributed axon models with both conventional and modified cable equations. `results = main_ES_HH(mod_prmtr, out_ctrl)`. Call function in a loop (parallelized) or run on computer cluster to cover all parameters `id` (see below). Input arguments:
    - `mod_prmtr`: structure specifying model parameter. Fields:
      - `model_name`, a string for the model: `'UF_TrmB_HH'`, `'UF_TrmB_RMG'`, `'UF_TrmS_RMG'`, `'UF_BndO_HH'`, `'UF_BndA_HH'`, `'UF_BndO_RMG'`, `'UF_BndA_RMG'`, `'PE_A_HH'`, `'DE_A_HH'`, `'PE_A_RMG'`, and `'DE_A_RMG'`.
      - `id`: integer, parameter ID. Range for the: `UF`: 1-600. `PE` and `DE`: 1-475.
    - `out_ctrl`: structure specifying outputs. Same as above for **`main_SC_HH.m`**.

  Output arguments:

        - `results`: structure containing the simulation results. Fields:
            - `th_CE`: threshold obtained with the conventional cable equation.
            - `th_MCE`: threshold obtained with the modified cable equation.
            - `th_per_diff_MCE`: percentage different of thresholds comparing `th_MCE` versus `th_CE`.
    - **specify_model_SC_HH.m**, **specify_model_ES_HH.m**, and **specify_model_ES_RMG.m**: specifies model, generates cable and stimulation waveform, calculates E-field, and sets parameters for solver and threshold search. Called by **main_SC_HH.m**, **main_ES_HH.m** and **main_ES_RMG.m**, respectively.
    - **simulate_cable_SC.m**, **simulate_cable_HH.m**, and **simulate_cable_RMG.m**: runs simulation, returns whether stimulation is suprathreshold or subthreshold. Called by **threshold_finding.m** (located in subfolder).
    - **HH.m**: ion channel gating variables. Called by **simulate_SC_HH.m** and **simulate_cable_HH.m**.
    - **RMG.m**: ion channel gating variables. Called by **simulate_cable_RMG.m**.
    - **compile_all.m**: compile data from individual simulations for all models. Generates `SOURCE_TARGET_MEMBRANE`**_compiled_result.mat** in each of the subfolders. Run after all simulations complete.
    - **plot_UF_Trm_combined.m**, **plot_UF_Bnd_combined.m**, **plot_PE_DE_HH_combined.m**, **plot_PE_DE_RMG_combined.m**: generates **Fig. 6–9** of the paper. Only .tif format provided; .fig files can be regenerated by running the scripts.
- **Shared functions and data**: folder containing code and data shared for both axon models.
    - **create_folders.m**: creates subfolders for saving data, logs, and figures. Called by **main_SC_HH.m**, **main_ES_HH.m**, and **main_ES_RMG.m**.
    - **write_fun.m**: output function for writing log files or displaying in MATLAB command lines. Called by many functions.
    - **threshold_finding.m**: function to find threshold for a specific model set-up. Called by **main_SC_HH.m**, **main_ES_HH.m**, and **main_ES_RMG.m**, and calls **simulate_SC_HH.m**, **simulate_cable_HH.m**, or **simulate_cable_RMG.m** various times, respectively.
    - **tridiag.m**: solves backward Euler of the cable equation using the tridiagonal matrix algorithm for linear neuronal topology. See solver descript below. Called by **simulate_SC_HH.m**, **simulate_cable_HH.m**, and **simulate_cable_RMG.m** for each backward Euler step.
- **Plot functions**: folder containing code and figure related to visualization for each model, and additional function for the combined plot functions. Only .tif format provided; .fig files can be regenerated by running the scripts
- **UF_Axon_HH**, **UF_Soma_HH**, **UF_TrmB_HH**, **UF_TrmB_RMG**, **UF_TrmS_RMG**, **UF_BndO_HH**, **UF_BndA_HH**, **UF_BndO_RMG**, **UF_BndA_RMG**, **PE_A_HH**, **DE_A_HH**, **PE_A_RMG**, and **DE_A_RMG**: Folders containing simulation results and outputs.
    - **Results**: subfolder containing simulation results generated by **main_SC_HH.m**, **main_ES_HH.m**, or **main_MS_RMG.m** and saved in **results_**`id`**.mat**.

    - SOURCE_TARGET_MEMBRANE**_compiled_result.mat**: all simulation results for the given model set-up compiled from the **Results** subfolder by **compile_all.m**.
    - **Logs**: subfolder containing log files generated by **main_SC_HH.m**, **main_ES_HH.m**, or **main_MS_RMG.m** documenting the threshold search process in **log_**id**.txt**.
    - **Figures**: subfolder for figures. Examples figures provided in .tif format. Original .fig files not included due to the file size (total of tens of GBs for all simulations) but can be generated and examined by re-running the simulations of interest (typically between a few seconds to a few hours depending on parameter).
- **PE_A_HH.slurm**: Example slurm script to run **main_ES_HH.m** on a computer cluster for model 'PE_A_HH' (all parameters id 1-475, with all 3 types of outputs, i.e., .mat file, .txt log, and .fig figure).