Multiarea Hill-Tononi thalamocortical network model 2.4

This is a work in progress. The final model will be released in SGA2 M24.

Changes since Ver. 2.3

(1) It was discovered that in sleep, all cortical excitatory cells showed intrinsic bursting (IB) activity in response to a step current input. The intention was to have only 30% of layer 5/6 (L56) excitatory cells configured as IB cells. This was remedied by adjusting the parameter tau_D_KNa, which determines the decay of the Na/Ca-activated K^+ conductance. Specifically, tau_D_KNa = 700 ms in regular spiking excitatory cells (ExCort) and 1500 ms in IB cells (in both wake and sleep), The latter now have their own cell model (ExCortIB) which can be specified in a state file (e.g. wake_parameters_2.4.txt). The fraction of L56 excitatory cells that are IB cells is specified in network_parameters_2.4.txt. Some tuning of g_peak_NaP in thalamic cells was also necessary to achieve acceptable firing rates.

(2) Although regular spiking ExCort cells do not exhibit IB activity in response to current steps, in sleep they do exhibit bursting in response to Poisson synaptic input. Because of this, with sufficient synaptic input during sleep (e.g. as specified in the suppplied sleep_parameters_2.4.txt) the network will still exhibit slow oscillations in the absence of IB cells (set `Layer_5/6_burster_fraction` to 0 in network_parameters_2.4.txt). On the other hand, setting g_NaL = 0.2 in ExCortIB cells in sleep will make these cells intrinsically active (i.e. active in the absence of any input) so that the network will exhibit slow oscillations in the absence of synaptic input. Thus we now have two kinds of sleep networks: (i) IB cells with no intrinsic activity; (ii) IB cells with intrinsic activity due to enhanced g_NaL. The first of these networks requires synaptic input to be active; the second doesn't. The behaviour of these networks is qualitatively similar.

(3) LFP data are now saved as centralised values for TMS rather than *z* scores because the latter are not appropriate for comparing TMS responses of different states with differing LFP SDs. Otherwise LFP data are saved as the raw values.

(4) Some additional parameters can now be set in network_parameters_2.4.txt, and some additional information is saved to output.txt.

Model description

We have implemented in NEST a 'toy brain' comprising left (L) and right (R) hemispheres, each with three cortical areas (C1-C3) and associated thalamic and reticular nuclei. Each neuronal layer comprises hybrid conductance-based/IAF neurons based on Hill and Tononi (2005). Intrahemispheric connectivities are based on Esser, Hill et al. (2009), with modifications to forward synaptic weights to enhance the propagation of the response to a simulated TMS pulse. Homotopic and heterotopic interhemispheric connectivities are based on data from various sources (Wise and Jones 1976, Douglas and Martin 2004, Petreanu, Huber et al. 2007, Harris and Shepherd 2015). To access the files described below, download and extract the file multiarea_ht_model_2.4.zip.

In order to run the model, copy the files in folder nest_ht_neuron_models to the NEST folder "models" before compiling NEST. The model is run with the python script Run_multiarea_HT_network_2.4.py. This calls the scripts Build_multiarea_HT_network_2.4.py and Simulate_multiarea_HT_network_2.4.py. The first of these builds the network and adds recorders and any synaptic bombardment. The second controls the simulation and saves the recorded data to an output folder. The files Functions_multiarea_HT_network_2.4.py and User_functions_multiarea_HT_network_2.4.py must also be present.

An example of spontaneous activity during a wake-to-sleep transition is in folder output_wake-sleep_2.4, while simulated TMS data are in output_wake_TMS_2.4 and output_sleep_TMS_2.4. Subfolder names begin with "L" or "R" followed by:

C#L4 – cortical layer 4
C#L23 – cortical layers 2/3
C#L56 – cortical layers 5/6
R# – reticular thalamic nucleus
TC# – thalamic core cells
TM# – thalamic matrix cells
TI# – thalamic inhibitory cells

with # = 1-3, indicating the brain area. Each subfolder contains excitatory and inhibitory cell spike times in files ...Ex_sp.txt and ...In_sp.txt, as well as membrane potential (V_{m}) data in files ..._V_m.txt. The latter contains mean V_{m} data for the layer as well as V_{m} for some randomly chosen cells. Spike times for these cells are in the ...Ex_Vm_sp.txt and ... In_Vm_sp.txt files. Mean cellular firing rates per sampling interval are in files …ExFR.txt and …InFR.txt, while topographic data for V_{m} are in files …_V_m_top.txt. The files ..._ ExIsynSum.txt contain the sum of synaptic currents for excitatory cells. The latter are used to calculate three proxies of LFP as described by Mazzoni, Linden et al. (2015): (1) $-I_{\mathrm{GABA}} = -I_{\mathrm{GABA\_A}}-I_{\mathrm{GABA\_B}}$; (2) $\Delta I = I_{\mathrm{AMPA}}+I_{\mathrm{NMDA}}-I_{\mathrm{GABA\_A}}-I_{\mathrm{GABA\_B}}$; (3) $RWS = I_{\mathrm{AMPA}}(t-\tau)+I_{\mathrm{NMDA}}(t-\tau)-\alpha(I_{\mathrm{GABA\_A}}+I_{\mathrm{GABA\_B}})$, where $\tau = 6$ ms and $\alpha = 1.65$. These proxies are saved to ...LFP.txt files as centralised data, i.e. $y-\bar{y}$ where y is the proxy. Average firing rates are in Pooled_firing_rates.txt and some miscellaneous data will be found in output.txt.

Various input files are required for the model to work. A *state*_parameters_2.4.txt file such as wake_parameters_2.4.txt (default) or sleep_parameters_2.4.txt is needed to define parameter values for model neurons, synaptic bombardment and synaptic depression; parameter names must not be changed. File neuron_layers_2.4.txt defines the neuron layers, specifying the number of rows and columns and the number of each type of neuron model to be placed at a node. Where the latter is a noninteger value, it is interpreted as a proportion of the nodes, chosen randomly. The file network_connections_2.4.txt defines connectivities for excitatory (E) and inhibitory (I) cells in the various layers and areas. Connections are made probabilistically within a mask of radius using a Gaussian kernel with maximum probability *Pmax*. Based on inspection of Synthesis code (in which the original Hill-Tononi model was implemented) the Gaussian standard deviation is given by $\sigma = \sigma_0 + \beta r$, with $\sigma_0 = 0.308$ (set in network_parameters_2.4.txt) and $\beta = 0.616$. Synaptic weights (*Strength*) are also given, together with mean transmission delays in ms (*Mean_Delay*) and their standard deviations (*SD*). Other network parameters such as the proportion of layer 5/6 (L56) excitatory cells that have intrinsic bursting (IB) activity can be set in network_parameters_2.4.txt. The file simulation_parameters.txt may be used to control various simulation parameters:

`Threads: 4`	(number of local threads)
`RNG_seed: 123456`	(negative to select a ‘random’ seed)
`Resolution(ms): 0.25`	(iteration step size)
`Sampling_interval(ms): 1.0`	(for recording data)
`Simulation_interval(ms): 100.0`	(interval between data saves)
`Rebuild_network(yes/no): yes`	(for multiple simulations; see below)
`State_file(_parameters_2.4.txt):`	(e.g. “wake” (default) or “sleep”)
`Settling_time(s): 1.0`	(no data recorded during this time)
`Run_time(s): 3.0`	(settling time excluded)
`Record_all_spike_times: yes`	(“yes” for ..Ex_sp.txt and ...In_sp.txt files)
`Time_series_for_variables: V_m`	(NEST ht_neuron model recordables*)
`Number_of_cells_for_time_series: 3`	(for ..._V_m.txt, …_g_AMPA etc. files)
`Topographics_for_V_m(yes/no): yes`	(generate …_V_m_top.txt files)
`Topographics_for_layers: LC2L23 LC2L4 LC2L56`	
`Topographic_sampling_interval(ms): 10.0`	
`TMS(yes/no): no`	(toggle TMS)
`TMS_target: LC1L23 LC1L4 LC1L56`	(target layers for TMS pulse)
`TMS_activates_interneurons(yes/no): yes`	
`Number_of_TMS_runs: 3`	(TMS pulses will be evenly distributed
`On_time(s): 1.0`	between On_time and Off_time. Also used
`Off_time(s): 2.0`	with neuron_parameters_2.4.txt; see below.)
`Antialias_filter(yes/no): no`	(Filter at less than half the sampling rate
`Filter_type(butter/bessel): butter`	to avoid aliasing (if this is a problem) but at
`Filter_cutoff_frequency_(Hz): 300`	the cost of increased memory usage.)
`Input_folder:`	(default = current working folder)
`state_parameters_2.4.txt_from_input_folder(yes/no): yes`	
`network_parameters_2.4.txt_from_input_folder(yes/no): yes`	
`neuron_layers_2.4.txt_from_input_folder(yes/no): yes`	
`network_connections_2.4.txt_from_input_folder(yes/no): yes`	
`neuron_parameters_2.4.txt_from_input_folder(yes/no): yes`	
`User_functions...2.4.txt_from_input_folder(yes/no): yes`	
`Output_folder_(relative_to_input_folder):`	(default = “output”)
`Copy_input_files_to_output_folder(yes/no): yes`	
`Delete_output_folder_if_already_present(yes/no): yes`	
`Compress_output_folder(yes/no): no`	

*V_m g_AMPA g_NMDA g_GABA_A g_GABA_B I_NaP I_KNa I_h I_T theta
(space-delimited as shown here; or choose “all”).

The input folder may be specified relative to the current working folder or as an absolute path. It may also be passed as a command line argument to Run_multiarea_HT_network_2.4.py (without quotes).

In the latter case simulation_parameters_2.4.txt will be imported from the input folder; otherwise it is always loaded from the current working folder. Alternatively give the name of a file ending with .txt containing a list of input folders (one per line) to perform several simulations consecutively.

If TMS is selected, TMS pulse times will be saved to TMS_times.txt and recorded data will be saved in folders output_1, output_2... A TMS pulse is simulated by the simultaneous firing of cells in the target area (Pashut, Magidov et al. 2014).

If antialiasing is selected, a sampling interval of `Resolution(ms)` will be used. Then at the end of a simulation interval the resulting time series will be low pass filtered and downsampled to a sampling interval of `Sampling_interval(ms)`. Because of increased memory usage, it may be necessary to reduce `Sampling_interval(ms)` when using an antialias filter.

A multiplier for the connection mask diameter can be set in network_parameters_2.4.txt; the values of σ_0, τ and α may also be set in this file. The file neuron_parameters_2.4.txt is used to change neuron model parameters during a simulation. Here is an example:

```
Layer    Model    Parameter      Values
*        *        g_NaL            0.13     0.15
*        *        g_KL             1.0      1.6
C        *        g_peak_NaP       0.5      1.0
T        Ex       g_peak_NaP       0.35     1.0
T        In       g_peak_NaP       0.5      1.0
LR       *        g_peak_NaP       0.275    0.35
RR       *        g_peak_NaP       0.275    0.35
LC       *        g_peak_AMPA      0.025    0.0375
RC       *        g_peak_AMPA      0.025    0.0375
LC       *        g_peak_KNa       0.5      2.0
RC       *        g_peak_KNa       0.5      2.0
C        Ex       CEx_bomb_rate    500.0    0.0
C        In       CIn_bomb_rate    250.0    0.0
T        Ex       TEx_bomb_rate    500.0    0.0
T        In       TIn_bomb_rate    250.0    0.0
*        IB       g_NaL            0.13     0.2
```

The first line must be present as shown. “Parameter” may be an ht_neuron settable (such as those in wake_parameters_2.4.txt and sleep_parameters_2.4.txt) or a synaptic bombardment rate in Hz (CEx_bomb_rate, CIn_bomb_rate, TEx_bomb_rate, TIn_bomb_rate; C = L4 cortex, T = thalamus). “Layer” must be a string that identifies one or more of the layers in Neuron_layers_2.4.txt. Similarly “Model” must be a string that identifies one or more of the models in wake_parameters_2.4.txt and sleep_parameters_2.4.txt. You may also enter “*” if all layers or models are to be targeted for parameter changes. If a layer or model is specified more than once, the last specification takes precedence. “Values” may contain one value (v_1) or two values (v_1 and v_2). v_1 sets the parameter value at the start of the simulation (t = 0). If v_2 is present, the parameter will be ramped from v_1 to v_2

over the time interval On_time to Off_time (actually 'staircased' with a time step equal to the simulation interval). If Off_time = Run_time then a step change from v_1 to v_2 will occur at t = On_time. If On_time = Run_time then no change in parameter values will occur. The above example causes a transition from wake to sleep (compare parameter values in wake_parameters_2.4.txt and sleep_parameters_2.4.txt). In this particular example no synaptic input is needed in sleep because the IB cells are intrinsically active due to the increase in g_NaL. If the line including "IB" is omitted, the network will be silent in sleep, unless the second set of synaptic bombardment rates are set to positive values (at least for the excitatory neurons). Note that parameter changes specified in neuron_parameters_2.4.txt and TMS are not mutually exclusive, so make sure you don't have both active unless that is your intention.

The file User_functions_multiarea_HT_network_2.4.py contains functions that, as supplied, do nothing. Of course the user may insert his/her own code to change parameters and/or analyse results. Multiple simulations may also be run using different parameter values. The use of these functions is described in User_functions_multiarea_HT_network_2.4.py.

Simulation results

Example output is in folders output_wake-sleep_2.4, output_wake_TMS_2.4. and output_sleep_TMS_2.4.

References

Douglas, R. J. and K. A. C. Martin (2004). "NEURONAL CIRCUITS OF THE NEOCORTEX." *Annual Review of Neuroscience* **27**(1): 419-451.
Esser, S. K., S. Hill and G. Tononi (2009). "Breakdown of Effective Connectivity During Slow Wave Sleep: Investigating the Mechanism Underlying a Cortical Gate Using Large-Scale Modeling." *Journal of Neurophysiology* **102**(4): 2096-2111.
Harris, K. D. and G. M. G. Shepherd (2015). "The neocortical circuit: themes and variations." *Nature Neuroscience* **18**(2): 170-181.
Hill, S. and G. Tononi (2005). "Modeling, sleep and wakefulness in the thelamocortical system." *Journal of Neurophysiology* **93**(3): 1671-1698.
Mazzoni, A., H. Linden, H. Cuntz, A. Lansner, S. Panzeri and G. T. Einevoll (2015). "Computing the Local Field Potential (LFP) from Integrate-and-Fire Network Models." *Plos Computational Biology* **11**(12).
Pashut, T., D. Magidov, H. Ben-Porat, S. Wolfus, A. Friedman, E. Perel, M. Lavidor, I. Bar-Gad, Y. Yeshurun and A. Korngreen (2014). "Patch-clamp recordings of rat neurons from acute brain slices of the somatosensory cortex during magnetic stimulation." *Frontiers in Cellular Neuroscience* **8**: Artcle 145.
Petreanu, L., D. Huber, A. Sobczyk and K. Svoboda (2007). "Channelrhodopsin-2-assisted circuit mapping of long-range callosal projections." *Nature Neuroscience* **10**(5): 663-668.
Wise, S. P. and E. G. Jones (1976). "The organization and postnatal development of the commissural projection of the rat somatic sensory cortex." *Journal of Comparative Neurology* **168**(3): 313-343.