Real-Time Simulation of Large-Scale Neural Models using NCS

Laurence C. Jayet Bray Roger V. Hoang Emily R. Barker Frederick C. Harris, Jr.



Organization for Computational Neurosciences 2012

Saturday, July 21



Today's Outline

• First Hour

- o Introduction
- o Equations and Implementation
- o Requirements and Simulation on a Single Machine
- o Input Language

Second Hour

- o Simple Model
- o Parameters Presentation and Testing
- o Output Analysis

Third Hour

- o Simulation on Multiple Machines
- o Software Tools
- o Robotic System Configuration
- o Larger Networks and Complete Loop Execution
- o Future Directions and Summary

Today's Outline

• First Hour

- o Introduction
- o Equations and Implementation
- o Requirements and Simulation on a Single Machine
- o Input Language

Second Hour

- o Simple Model
- Parameters Presentation and Testing
- o Output Analysis

• Third Hour

- o Simulation on Multiple Machines
- o Software Tools
- o Robotic System Configuration
- o Larger Networks and Complete Loop Execution
- o Future Directions and Summary

Introduction

Presenters

- Fred Harris
 - Professor Department of Computer Science and Engineering
 - Ph.D. in Computer Science
- Laurence Bray
 - o Post Doctoral Research Associate,
 - Soon to be promoted to Research Assistant Professor
 - Ph.D. in Biomedical Engineering
- Roger Hoang
 - o Ph.D. Candidate
 - Planning to graduate this coming school year
 - Majoring in Computer Science
- Emily Barker
 - o B.S. Student
 - Starting her Senior year this fall
 - Majoring in Neuroscience



Reno, Nevada



University of Nevada, Reno



Brain Computation Lab

http://www.cse.unr.edu/brain/



Brain Computation Lab

Navigation

- Research Projects
- People
- Publications
- Sponsors
- Conferences
- Opportunities
- University of Nevada, Reno
- Department of Computer
 Science and Engineering
- School of Medicine
- Biomedical Engineering Program



Welcome to the Brain Laboratory!

Good Afternoon!

Founded in 2001, the brain lab is a joint research center between the departments of Computer Science & Engineering, Medicine, Physiology & Cell Biology, and the program of Biomedical Engineering. It also has neurobiological collaborations with the Brain Mind Institute at the EPFL (Switzerland), the University of Cergy Pontoise (France), and the University of Bonn (Germany).

Our researchers consists primarily of undergraduate/graduate students and alumni of the University of Nevada, Reno. They are actively developing computational innovations to understand the physiological processes that give rise to neocortical memory, learning, and cognition. Our models and experiments help understand brain pathophysiology and create brain-like artificial intelligence and neural prosthetic devices.

New Publications

- Design and Implementation of an NCS-NeuroML Translator
- Real-Time Human-Robot Interaction Underlying Neurorobotic Trust and Intent Recognition
- Correlation Maps Allow Neuronal Electrical Properties to be Predicted from Singlecell Gene Expression Profiles in Rat Neocortex
- Heterogeneity in the Pyramidal Network of the Medial Prefrontal Cortex

NCS History

• Version 1:1999

- o Matlab Goodman, Markram, and McKenna
- o 160-cell, 2-column architecture
 - Each cell was modeled as a single integrative compartment (point neuron) with a spike mechanism,
 - o calcium-dependent (AHP) channels, and
 - o voltage-sensitive A and M (muscarinic) potassium channels

• Version 1b: 1999

- o Direct translation to C from Matlab
- o 24 times faster.
- o tested on mixed excitatory-inhibitory networks of up to 1,000 cells

• Version 2: 1999

- code was then redesigned and rewritten for distributed processing on an existing 20-cpu cluster (Pentium II).
- o Initial trials of this code were performed on cortical networks of 2 to 1,000 cells

M.M. Kellog, H.R. Wills, and P.H. Goodman. "A biologically realistic computer model of neocortical associative learning for the study of aging and dementia." J. Investig. Med., 47(2), February 1999.

NCS History

• Version 3: 2001

- completely redesigned using object-oriented design principles and recoded in C++
- o bjects, such as cells, compartments, channels, and the like, model the corresponding cortical entities.
- The cells, in turn, communicate via messages passed through synapse objects.
- Input parameters allow the user to create many variations of the basic objects, in order to model measured or hypothesized biological properties.

E. Courtenay Wilson, Phillip H. Goodman, and Frederick C. Harris, Jr. "Implementation of a biologically realistic parallel neocortical-neural network simulator" in Proceedings of the 10th SIAM Conf. on Parallel Process. for Sci. Computing, Portsmouth, Virginia, March 2001.

NCS History



E. Courtenay Wilson, Frederick C. Harris, Jr., and Phillip H. Goodman. "A large-scale biologically realistic cortical simulator" in Proceedings of SC 2001, Denver, Colorado, November 2001

Code Optimization & Revisions

- Rewrote the input parser
- Worked on code base
 - o sevenfold sequential speedup over the version 3 code
 - added new features while shrinking our code base by more than 25%.
- Added More Biological Parameters.
- 35,000 cells and approximately 6.1 million synapses using 72% of the available 4GB of memory per node.

Code Optimization

Item	NCS3	NCS5	Ratio		
$Overhead^a$	294.167	1.897	155.1		
Base $\operatorname{Cell}/\operatorname{Cmp}^b$	0.020	3.035	153.6		
$Channel^{b}$	0.152	0.398	2.6		
$\operatorname{Report}^{c}$	0.017	4.113	239.4		
Synapse, 0 Hebb ^b	0.031	0.383	12.5		
Synapse, $+$ -Hebb ^b	0.020	0.368	18.1		
a) Seconds.					
b) Millions of Objects Processed per Second					

c) Millions of Values Reported per Second

Code Optimization



James Frye, James G. King, Christine J. Wilson, and Frederick C. Harris, Jr. "QQ: Nanoscale timing and profiling" In Proceedings of PMEO-PDS, Denver, CO, April 3-8 2005.

Hardware



2001

2002





P IV Xeon 2.2GHz (68 CPUs)



Myrinet 2000

PIII 1GHz (60 CPUs) ONR DURIP 2002: N000140210557 ONR DURIP 2001: N000140110552

Hardware



Sun v20z Opteron (60 CPUs)

6 118

ONR DURIP 2007:

Sun 4600s and 4500s 16 core boxes with 200GB of RAM connected by Infiniband And several 24TB disk arrays

ONR DURIP 2008:

Current NCS version 6

- GPU/CPU/cluster-based
- Multiple neuron types (version 5 + izhikevich + framework for others)
- Ability for multi-scale modeling

Current Hardware

GeForce GTX 480 Fastest GPU in the World

Memory	1536MB / 384-bit GDDR5	
Cores	480	
Gfx / Proc / Mem Clock	700 / 1401 / 1848 MHz	
Power Connectors	6-pin + 8-pin	
Power	250W	
SLI	3-way	
Length	10.5 inches	
Thermal	Dual Slot Fansink	
Outputs	DL-DVI DL-DVI mini-HDMI	



GeForce GTX 690 Specifications

CUDA Cores	3072
Base Clock	915 MHz
Boost Clock	1019 MHz
Memory Config	4GB / 512-bit GDDR5
Memory Speed	6.0 Gbps
Power Connectors	8-pin + 8-pin
TDP	300W
Outputs	3x DL-DVI Mini-Displayport 1.2
Bus Interface	PCI Express 3.0

Current Optimizations

- C++11
- Heavily threaded
 - o Latency hiding
 - o Increased occupancy
- Modular message passing design
- GPU usage for parallel computation
- Load-balancing across heterogeneous clusters

Current Optimization

Cell Count	Synapse Count	NCS5 Simulation Time (Sec)	NCS6 Simulation Time (Sec)
~1,000	~2700	3.3	<1
~10,000	~250,000	4.4	<1
~100,000	~25,000,000	99.0	1.1

Comparison with other Simulators

Advantages:

- No programing language experience
- o Large-scale networks simulation
- o Real-time Execution
- o Good for behavior, systems, and networks
- o Framework for different level of abstraction

Disadvantages

- o Lack of cellular and subcellular details
- o No anatomical visualization yet

Romain Brette, Michelle Rudolph, Ted Carnevale, Michael Hines, David Beeman, James M. Bower, Markus Diesmann, Abigail Morrison, Philip H. Goodman, Frederick C. Harris, Jr., Milind Zirpe, Thomas Natschlager, Dejan Pecevski, Bard Ermentrout, Mikael Djurfeldt, Anders Lansner, Olivier Rochel, Thierry Vieville, Eilif Muller, Andrew P. Davison, Sami El Boustani and Alain Destexhe "Simulation of networks of spiking neurons: A review of tools and strategies" Journal of Computational Neuroscience December 2007 (Vol 23), pp 349-398. Equations and Implementation

```
int refractoryTime = refractoryTime[index];
float calcium = oldCalcium[index];
if (refractoryTime >= 0)
    newVoltage[index] = spikeShapes[index][refractoryTime];
    --refractoryTime;
    refractoryTime[index] = refractoryTime;
   newCalcium[index] = oldCalcium[index];
else
    float I = 0.0f;
    float voltage = oldVoltage[index];
   I += inputCurrent[index];
   I += channelCurrent[index];
   I -= leakConductance[index] * (voltage - leakReversal[index]);
   I += synapticCurrent[index];
    if (bit::extract(voltageClamp, index))
       voltage = clampVoltage[index];
    else
        float restVoltage = restVoltage[index];
       voltage = restVoltage +
                  (voltage - restVoltage) * persistence[index] +
                  dtC[index] * I;
    voltage = voltage < -80.0f? -80.0f : voltage;</pre>
    //Check for firing
    if (voltage > threshold[index])
       refractoryTime = spikeShapeLength[index] - 1;
       voltage = spikeShapes[index][refractoryTime];
       _refractoryTime[index] = refractoryTime - 1;
       warpResult = mask;
        calcium += caSpikeIncrement[index];
    calcium *= caPersistence[index];
    newCalcium[index] = calcium;
    newVoltage[index] = voltage;
```

Compartments

```
unsigned int compartmentID = compartmentIDs[index];
float compartmentVoltage = compartmentVoltages[compartmentID];
unsigned int tauMIndex = tauMIndices[index];
unsigned int tauMEnd = tauMIndices[index + 1];
float t m = 0.0f;
for (; tauMIndex < tauMEnd; ++tauMIndex)</pre>
    if (compartmentVoltage < tauMVoltage[tauMIndex])</pre>
        t m = tauMValue[tauMIndex];
        break;
float m oo =
   1.0f / (1.0f + exp(-(compartmentVoltage - eHalfMinM[index]) /
                       slopeFactorM[index]));
float m = m[index];
m += (m \circ o - m) * dt / t m;
m = (m < 0.0f)? 0.0f : (m > 1.0f)? 1.0f : m;
m[index] = m;
unsigned int tauHIndex = tauHIndices[index];
unsigned int tauHEnd = tauHIndices[index + 1];
float t h = 0.0f;
for (; tauHIndex < tauHEnd; ++tauHIndex)</pre>
    if (compartmentVoltage < tauHVoltage[tauHIndex])</pre>
        t h = tauHValue[tauHIndex];
float h oo =
   1.0f / (1.0f + exp((compartmentVoltage - eHalfMinH[index]) /
                        slopeFactorH[index]));
float h = h[index];
h += (h \circ o - h) * dt / t h;
h = (h < 0.0f)? 0.0f : (h > 1.0f)? 1.0f : h;
h[index] = h;
float I = unitaryG[index] * strength[index] * pow(m, mPower[index]) *
          pow(h, hPower[index]) *
          (reversalPotential[index] - compartmentVoltage);
atomicAdd(compartmentCurrents + compartmentID, I);
```

Ka Channels

Kahp Channels

Km Channels

```
unsigned int index = indices[block::thread()];
const unsigned char type = RSEType[index];
float USE = USE[index];
float RSE = RSE[index];
float USEBase = USEBase[index];
float dt = elapsedTime - lastFireTime[index];
float firingValue = maxG[index];
if (type & 0x2u) //Facilitation
    USE += (1.0f - USEBase) * USE *
           exp(1000.0f * -dt / tauFacilitation[index]);
    USE = math::clamp(USE, 0.0f, 1.0f);
    USE[index] = USE;
if (type & 0x1u) //Depression
    RSE = 1.0f + ((RSE * (1.0f - USEBase) - 1.0f) *
                  exp(1000.0f * -dt / tauDepression[index]));
    RSE = math::clamp(RSE, 0.0f, 1.0f);
    RSE[index] = RSE;
ι
switch(type)
case 0x0: //None
    firingValue *= USEBase;
break;
case 0x1: //Depression
    firingValue *= USEBase * RSE;
break;
case 0x2: //Facilitation
    firingValue *= USE;
break;
case 0x3: //Both
    firingValue *= USE * RSE;
break;
default:
break;
```

Short-Term Learning

//Negative learning

```
if (learningOn[index])
```

```
float postDT = elapsedTime - lastPostFireTime[index];
switch(learningType[index])
case 0: //None
break;
case 1: //Exponential
    float positiveLearningModulator = positiveLearningModulator[index];
    positiveLearningModulator *=
        exp(1000.0f * -dt / positivePeakTime[index]);
    positiveLearningModulator += positivePeakDeltaUSE[index];
    positiveLearningModulator[index] = positiveLearningModulator;
    USEBase -=
        exp(1000.0f * -postDT / negativePeakTime[index]) *
        negativeLearningModulator[index];
    if (USEBase < 0.0f) USEBase = 0.0f;</pre>
    USEBase[index] = USEBase;
break;
case 2: //Triangle
    float negativeWindowWidth = negativeWindowWidth[index];
    if (postDT < negativeWindowWidth)</pre>
        float peakTime = negativePeakTime[index];
        float dUSE = negativePeakDeltaUSE[index];
        if (postDT < peakTime)</pre>
            dUSE *= (postDT / peakTime);
        else
            dUSE *= 1.0f - (postDT - peakTime) /
                    ( negativeWindowWidth - peakTime);
        USEBase -= dUSE;
        if (USEBase < 0.0f) USEBase = 0.0f;</pre>
        USEBase[index] = USEBase;
break;
default:
break;
```

Long-Term Negative Learning

```
float USEBase = USEBase[index];
 float preDT = elapsedTime - lastPreFireTime[index];
 switch(learningType[index])
 case 0: //None
 break;
 case 1: //Exponential
     float postDT = elapsedTime - lastPostFireTime[index];
     float negativeLearningModulator = negativeLearningModulator[index];
     negativeLearningModulator *=
         exp(1000.0f * -postDT / negativePeakTime[index]);
     negativeLearningModulator += negativePeakDeltaUSE[index];
      negativeLearningModulator[index] = negativeLearningModulator;
     USEBase +=
         exp(1000.0f * -preDT / positivePeakTime[index]) *
         positiveLearningModulator[index];
     if (USEBase > 1.0f) USEBase = 1.0f;
      USEBase[index] = USEBase;
 break;
 case 2: //Triangle
     float positiveWindowWidth = positiveWindowWidth[index];
     if (preDT < positiveWindowWidth)</pre>
         float peakTime = positivePeakTime[index];
         float dUSE = positivePeakDeltaUSE[index];
         if (preDT < peakTime)</pre>
             dUSE *= (preDT / peakTime);
             dUSE *= 1.0f - (preDT - peakTime) /
                      ( positiveWindowWidth - peakTime);
         USEBase += dUSE;
         if (USEBase > 1.0f) USEBase = 1.0f;
         USEBase[index] = USEBase;
 break;
 default:
 break;
//if learning on
```

Long-Term Positive Learning

```
unsigned int firingIndex = firingIndices[index];
unsigned int PSGCount = PSGCounts[index];
--PSGCount;
float PSGValue = PSGs[firingIndex][PSGCount];
unsigned int postNeuronID = postNeuronIDs[firingIndex];
float voltage = synapticReversal[firingIndex];
voltage -= voltages[postNeuronID];
float firingValue = firingValues[index];
float current = voltage * firingValue * PSGValue;
atomicAdd(synapticCurrents + postNeuronID, current);
bool save = false;
unsigned int saveIndex;
if (PSGCount > 0)
    save = true;
    saveIndex = atomicAdd(&numQueued, lu);
    if (saveIndex < block::size())</pre>
        gueuedIndices[saveIndex] = firingIndex;
        queuedPSGCounts[saveIndex] = PSGCount;
        queuedFiringValues[saveIndex] = firingValue;
        save = false;
    else
        saveIndex -= block::size();
```

Post Synaptic Conductance

NCS 6 Implementation

- Plugin interface for multiple model support
 - o Currently have:
 - NCS 5 LIF Neurons
 - Izhikevich Neurons
 - o ability to design your own
 - o Have a student working on a Neuron CPU plugin.
- Runs on CPUs, CUDA devices, and OpenCL devices simultaneously

Requirements

NCS5 Software / Hardware

Linux based operating system

NCS6 Software / Hardware

Linux based operating system

• NVDIA GPUs

NCS5- Packages Needed

- bison
- flex
- mpi-run

- : sudo apt-get install bison
- : sudo apt-get install flex
- : sudo apt-get
 install openmpi-bin
 : sudo apt-get install
 openmpi-dev

NCS6- Packages Needed

- bison : sudo apt-get install bison
- cmake : sudo apt-get install cmake
- cuda toolkit
- : http://developer.nvidia.com/
 - o cd /home/userName/Downloads
 - sh <cuda_toolkit_installer_name>
- doxygen
- : sudo apt-get install doxygen
- flex : sudo apt-get install flex
- g++ version 4.4 : sudo apt-get install g++-4.4
- g++ version 4.6+
- mercurial : sudo apt-get install mercurial
- mpi-run

: sudo apt-get install mercurial : sudo apt-get install openmpi-bin

: sudo apt-get install openmpi-dev
Simulation on a single machine

NCS5 Steps

- To compile code:
 o Make
- After the code is compiled, you run NCS5 in the directory with the input file
- To run code:
 o ncs5pe <input file>

NCS6 Steps

- cd /home/userName/NCS6/NCS6/build
- To specify the number of devices available on the computer for the program (Only do this step once)
 - o mpirun applications/clusterSpecifier/clusterSpecifier single.cluster
 - o applications/clusterInfo/clusterInfo single.cluster

• To compile code:

applications/ncsDistributor/ncsDistributor <space>
 .../files/NCS6/folderName/fileName single.cluster ncsout

• To run code:

o applications/simulator/simulator/ ncsout



Input Language

- Define the simulation as a whole
- Preliminary outline of other structures
 - o Anatomy
 - o Stimuli
 - o Reports
- Extrinsic connections
- Include files

BRAIN	
TYPE	SIMPLE_MODEL_model
JOB	SIMPLE_MODEL_model
FSV	1e3
DURATION	1
SEED	-21
DISTANCE	NO
######################################	########## COLUMN TYPE####################################
######################################	######## STIM INJECT####################################
+++++++++++++++++++++++++++++++++++++++	######## REPORTS ####################################
REPORT VOL TAGE	$\begin{array}{c} CFII & I \end{array}$
REPORT VOLTAGE	
END_BRAIN	



- Columns
- Layers
- Cells
- Compartments
- Channels

Anatomy

#### COLU	######################################	*######################################	Define	Column	Shell	s ####################################
	TYPE	SIMPLE_MODEL_	SHELL			
	HEIGHT	800				
	LOCATION	0	800			
END_	COLUMN_SHELL					
#### COLU	################ MN	*######################################	Fill Co	olumns	######	******
	ТҮРЕ	SIMPLE_MODEL_	COLUMN			
	COLUMN_SHELL	SIMPLE_MODEL_	SHELL			
END_	COLUMN	layer_SIMPLE_	MODEL			
#### LAYE	######################################	*######################################	Define	Layer	Shells	#######################################
	TYPE	layer_SIMPLE_	MODEL_s	shell		
	LOWER	Θ				
	UPPER	400				
END_	LATER_SHELL					
#### LAYE	############### R	*##############	Fill La	ayers #	#######	#######################################
	TYPE		layer_9	SIMPLE_	MODEL	
	LAYER_SHELL	-	layer_S	SIMPLE_	MODEL_	shell
					.1 .2	10
END	LAYER		STHEF	NODEL_	2	10

Stimulus

- External Stimulation (visual, audio...)
- Type of signals
 - o Linear
 - o Pulse
 - o Noise
 - o File-based
- Multiple times
- Different Destinations

Stimulus

#######################################	#### STIMULUS	INJECTS	#######################################	################	
STIMULUS_INJECT TYPE S STIM TYPE r	SIMPLE_MODEL_ST realstim SIMPLE	IM MODEL			
INJECT S	SIMPLE_MODEL_CO	LUMN layer	_SIMPLE_MODEL	SIMPLE_MODEL_1	somaE 1
END_STIMULUS_INJECT					
#######################################	#########defin	e STIMULUS	#######################################	################	
STIMULUS					
TYPE	re	alstim_SIMP	LE_MODEL		
MODE	CU	RRENT			
PATTERN	PU	LSE			
DYN_RANGE	0	75			
TIMING	EX	АСТ			
AMP_START	4				
WIDTH	. 0	10			
TIME_START	0.	500			
TIME_END	0.	600			
END_STIMULUS					

Connections

- Extrinsic and intrinsic connections
- Synapse connections
- From the source to the destination
- With or without decaying distance effects
- Recurrent connections

Connections

#########################	****
<pre># connections</pre>	
##########################	****
CONNECT	
	TWO_CELL_MODEL_COLUMN layer_TWO_CELL_MODEL TWO_CELL_MODEL_1 somaE
	TWO_CELL_MODEL_COLUMN layer_TWO_CELL_MODEL TWO_CELL_MODEL_2 somaE
	synEE_TWO_CELL_MODEL 1 0



- Connections between other cells and their compartments
- Excitatory
- Inhibitory
- Synaptic Waveform
- Learning
 - o Short term synaptic dynamics
 - Facilitation
 - Depression
 - o Long term synaptic dynamics (Hebbian Learning)
 - STDP rule

Synapses

###############SYNAPSES SIMPLE_M	MODEL_MODEL####################################
SYNAPSE TYPE synEE_SIMPLE_M SFD_LABEL NO_SFD LEARN_LABEL NO_STDP SYN_PSG PSGexcit MAX_CONDUCT 0.4 DELAY 0.005 0.010 SYN_REVERSAL 0 0 ABSOLUTE_USE 0.25 0.1 END_SYNAPSE	MODEL
######################################	TERM SYNAPTIC DYNAMICS ####################################
SYN_FACIL_DEPRESS TYPE SFD END_SYN_FACIL_DEPRESS	NO_SFD NONE
#######################################	LONG-TERM SYNAPTIC DYNAMICS ####################################
SYN_LEARNING TYPE LEARNING END_SYN_LEARNING	NO_STDP NONE
#######################################	SYNAPTIC CONDUCTANCE WAVEFORMS ####################################
SYN_PSG TYPE PSG_FILE END_SYN_PSG	<pre>PSGexcit ./input/EPSG_Vogels_FSV1k_TAU05.inc</pre>



Data about cells

• Report files:

- o Voltage
- o Current
- o Firecount
- o Channel
- o Synaptic strengths
- Automatically generated and saved

Reports

#######################################	SIMPLE_MODEL_MODEL REPORTS ####################################
REPORT	
TYPE	VOLTAGE CELL 1
CELLS	SIMPLE MODEL COLUMN layer SIMPLE MODEL SIMPLE MODEL 1 somaE
PROB	1
REPORT ON	VOLTAGE
FILENAME	SIMPLE MODEL 1 VOLTAGE E.txt
ASCII	
FREQUENCY	1
TIME_START	Θ
TIME_END	100
END_REPORT	
REPORT	
TYPE	VOLTAGE CELL 2
CELLS	SIMPLE_MODEL_COLUMN layer_SIMPLE_MODEL SIMPLE_MODEL_2 somaE
PROB	1
REPORT_ON	VOLTAGE
FILENAME	SIMPLE_MODEL_2_VOLTAGE_E.txt
ASCII	
FREQUENCY	1
TIME_START	Θ
TIME_END	100
END_REPORT	





Today's Outline

• First Hour

- o Introduction
- Equations and Implementation
- o Requirements and Simulation on a Single Machine
- o Input Language

Second Hour

- o Simple Model
- o Parameters Presentation and Testing
- o Output Analysis

• Third Hour

- o Simulation on Multiple Machines
- o Software Tools
- o Robotic System Configuration
- o Larger Networks and Complete Loop Execution
- o Future Directions and Summary

Simple Model





BRAIN	
TYPE	SIMPLE_MODEL_model
JOB	SIMPLE_MODEL_model
FSV	1e3
DURATION	1
SEED	-21
DISTANCE	NO
######################################	######################################
######################################	######## STIM INJECT####################################
+++++++++++++++++++++++++++++++++++++++	######## REPORTS ####################################
REPORT VOL TAGE	$\begin{array}{c} CFII & I \end{array}$
REPORT VOLTAGE	
END_BRAIN	

BRAIN	
TYPE	SIMPLE MODEL model
JOB	SIMPLE MODEL model
FSV	1e3
DURATION	1
SEED	-21
DISTANCE	NO
#######################################	######################################
COLUMN TYPE SIM	PLE MODEL COLUMN
#######################################	######### STIM INJECT####################################
STIMULUS INJECT	SIMPLE MODEL STIM
01110200_110201	
#######################################	######## REPORTS ####################################
REPORT VOLTAGE	
REPORT VOLTAGE	
NEI ONI VOETAGE	

BRAIN	
TYPE	SIMPLE MODEL model
JOB	SIMPLE MODEL model
FSV	1e3
DURATION	1
SEED	-21
DISTANCE	NO
############################	########## COLUMN TYPE####################################
COLUMN TYPE SIM	PLE MODEL COLUMN
#######################################	######### STIM INJECT####################################
STIMULUS INJECT	SIMPLE MODEL STIM
#######################################	######## REPORTS ####################################
REPORT VOLTAGE	
REPORT VOLTAGE	
NEI ONI VOLTAGE	

BRAIN	
TYPE	SIMPLE_MODEL_model
JOB	SIMPLE_MODEL_model
FSV	1e3
DURATION	1
SEED	-21
DISTANCE	NO
#############################	########## COLUMN TYPE####################################
COLUMN TYPE SIM	PLE MODEL COLUMN
—	
############################	######### STIM INJECT####################################
STIMULUS INJECT	SIMPLE MODEL STIM
—	
#######################################	######## REPORTS ####################################
REPORT VOLTAGE	CELL 1
REPORT VOLTAGE	CELL 2
END BRATN	

BRAIN	
TYPE	SIMPLE_MODEL_model
JOB	SIMPLE_MODEL_model
FSV	1e3
DURATION	1
SEED	-21
DISTANCE	NO
###########################	########### COLUMN TYPE####################################
COLUMN TYPE SI	MPLE MODEL COLUMN
_	
#############################	########## STIM INJECT####################################
STIMULUS INJECT	SIMPLE MODEL STIM
—	
#######################################	######### REPORTS ####################################
REPORT VOLTAG	E CELL 1
REPORT VOLTAG	E ^{CELL} 2
END BRAIN	
-	

TYPE	SIMPLE_MODEL_model
JOB	SIMPLE_MODEL_model
FSV	1e3
DURATION	1
SEED	-21
DISTANCE	NO
#####################	########## COLUMN TYPE####################################
OLUMN_TYPE SI	MPLE_MODEL_COLUMN
####################	########## STIM INJECT####################################
TIMULUS_INJECT	SIMPLE_MODEL_STIM
####################	######### REPORTS ####################################
EPORT VOLTAGE	E_CELL_1
EPORT VOLTAGI	E_CELL_2
RAIN	
	TYPE JOB FSV DURATION SEED DISTANCE ####################################

SIMPLE_MODEL_model					
SIMPLE MODEL model					
1e3					
1					
-21					
NO					
######### COLUMN TYPE####################################					
PLE_MODEL_COLUMN					
######### STIM INJECT####################################					
SIMPLE_MODEL_STIM					
######## REPORTS ####################################					
CELL_1					
CELL_2					

BRAIN						
TYPE	SIMPLE_MODEL_model					
JOB	SIMPLE_MODEL_model					
FSV	1e3					
DURATION	1					
SEED	-21					
DISTANCE	NO					
#######################################	######################################					
COLUMN TYPE S	SIMPLE MODEL COLUMN					
########################	######################################					
STIMULUS_INJECT	SIMPLE_MODEL_STIM					
######################################						
REPORT VOLTA	AGE_CELL_1					
REPORT VOLTA	AGE_CELL_2					
END_BRAIN						

BRAI	N	
	TYPE	SIMPLE_MODEL_model
	JOB	SIMPLE_MODEL_model
	FSV	1e3
	DURATION	1
	SEED	-21
	DISTANCE	NO
####	#####################	########### COLUMN TYPE####################################
	COLUMN_TYPE SI	1PLE_MODEL_COLUMN
####	#####################	########## STIM INJEC <mark>T####################################</mark>
ι ι	STIMULUS_INJECT	SIMPLE_MODEL_STIM
####	####################	######### REPORTS ####################################
	REPORT VOLTAGI	E_CELL_1
	REPORT VOLTAGI	E_CELL_2
END_	BRAIN	

BRA:	EN					
	TYPE	SIMPLE_MODEL_model				
	JOB	SIMPLE_MODEL_model				
	FSV	1e3				
	DURATION	1				
	SEED	-21				
	DISTANCE	NO				
####	\$#####################################	########## COLUMN TYPE####################################				
	COLUMN TYPE SI	MPLE MODEL COLUMN				
	_					
####	######################################					
	STIMULUS_INJECT	SIMPLE_MODEL_STIM				
####	<i><u>+</u>#######################</i> #############	#########_REPORTS ####################################				
	REPORT VOLTAG	E_CELL_1				
	REPORT VOLTAG	E CELL 2				
END	BRAIN					
	DIAIN					

Anatomy

######################################	Define Column Shells ###################################
TYPE SIMPLE_MODEL	_SHELL
HEIGHT 800	
LOCATION 0	800
END_COLUMN_SHELL	
######################################	Fill Columns ####################################
TYPE SIMPLE_MODEL	COLUMN
COLUMN_SHELL SIMPLE_MODEL	SHELL
END_COLUMN	_MODEL
######################################	Define Layer Shells ###################################
TYPE layer_SIMPLE	_MODEL_shell
LOWER 0	
UPPER 400	
END_LAYER_SHELL	
######################################	Fill Layers ####################################
TYPE	layer_SIMPLE_MODEL
LAYER_SHELL	layer_SIMPLE_MODEL_shell
	SIMPLE_MODEL_1 IO
	SINFLE_NOULL_Z IV

Anatomy

######################################	Define Column Shells
TYPE SIMPLE_MODEL WIDTH 300 HEIGHT 800	SHELL
LOCATION 0 END_COLUMN_SHELL	800
######################################	Fill Columns ####################################
TYPE SIMPLE_MODEL COLUMN_SHELL SIMPLE_MODEL LAYER_TYPE layer_SIMPLE END_COLUMN	_COLUMN _SHELL _MODEL
######################################	Define Layer Shells ###################################
TYPE layer_SIMPLE LOWER 0 UPPER 400	_MODEL_shell
LAYER	ritt Layers ####################################
TYPE LAYER_SHELL CELL_TYPE CELL_TYPE	layer_SIMPLE_MODEL layer_SIMPLE_MODEL_shell SIMPLE_MODEL_1 10 SIMPLE_MODEL_2 10
END_LAYER	

Anatomy

#### COLL	######################################	#######################################	Define	Column	Shells	#######################################
END	TYPE WIDTH HEIGHT LOCATION COLUMN SHELL	SIMPLE MODEL 300 800 0	SHELL 800			
#### COLU	######################################	############# SIMPLE_MODEL_ SIMPLE_MODEL_ layer_SIMPLE_	Fill Co COLUMN SHELL MODEL	olumns :	#######	*******
#### LAYE	######################################	############## layer_SIMPLE_ 0 400	Define MODEL_s	Layer shell	Shells	###############################
#### LAYE END_	######################################	*############# -	Fill La layer_S layer_S SIMPLE_ SIMPLE_	ayers # SIMPLE_ SIMPLE_ MODEL_ MODEL_	####### MODEL MODEL_s 1 1 2 1	######################################

Anatomy

####	!#####################################	#######################################	Define	Column	Shells	#######################################
COLU	MN_SHELL TYPE WIDTH	SIMPLE_MODEL_ 300	SHELL			
END_	HEIGHT LOCATION COLUMN_SHELL	800 0	800			
#### C0Ll	<i>!##############</i> JMN	#######################################	Fill Co	lumns #	#######	#######################################
END_	TYPE COLUMN_SHELL LAYER_TYPE COLUMN	SIMPLE_MODEL_ SIMPLE_MODEL_ layer_SIMPLE_	COLUMN SHELL MODEL			
####	######################################	#######################################	Define	Layer S	Shells	#######################################
	TYPE LOWER UPPER	layer_SIMPLE_ 0 400	_MODEL_s	hell		
END_	LAYER_SHELL					
#### Laye	<i>!###############</i> ER	<i>####################################</i>	Fill La	yers ##	#######	#######################################
END	TYPE LAYER_SHELI CELL_TYPE CELL_TYPE LAYER	L	layer_S layer_S SIMPLE_ SIMPLE_	SIMPLE_N SIMPLE_N MODEL_1 MODEL_2	40DEL 40DEL_s 1 1 2 1	hell 0 0
#### COLL	######################################	*######################################	Define	Column	Shells	;
--------------	--	---	--	--	--------------------------------	---
COLC	TYPE WIDTH	SIMPLE_MODEL_	SHELL			
END_	LOCATION COLUMN_SHELL	0	800			
#### COLU	######################################	*######################################	Fill Co	olumns a	#######	*****
END_	TYPE COLUMN_SHELL LAYER_TYPE COLUMN	SIMPLE_MODEL_ SIMPLE_MODEL_ layer_SIMPLE_	COLUMN SHELL MODEL			
####	######################################	+######################################	Define	Layer S	Shells	#######################################
	TYPE LOWER UPPER	layer_SIMPLE_ 0 400	MODEL_s	shell		
END_	LAYER_SHELL					
#### LAYE	######################################	+######################################	Fill La	ayers #	#######	*****
END_	TYPE LAYER_SHELI CELL_TYPE CELL_TYPE LAYER	-	layer_S layer_S SIMPLE_ SIMPLE_	SIMPLE_I SIMPLE_I MODEL MODEL	MODEL MODEL_s 1] 2]	hell .0 .0

####	######################################	*######################################	Define	Column	Shells	5 #####################################
CULU	TYPE	SIMPLE_MODEL_	SHELL			
	WIDTH HEIGHT	300 800				
	LOCATION	0	800			
END_	COLUMN_SHELL					
#### COLU	######################################	*#############	Fill Co	lumns a	#######	****
	ТҮРЕ	SIMPLE_MODEL_	COLUMN			
	COLUMN_SHELL	SIMPLE_MODEL_	SHELL			
END_	COLUMN	layer_SIMPLE_	MODEL			
#### LAYE	######################################	*######################################	Define	Layer	Shells	#######################################
	TYPE	layer_SIMPLE_	MODEL_s	hell		
	LOWER	0				
END_	LAYER_SHELL	400				
#### LAYE	######################################	*######################################	Fill La	iyers #	#######	****
	TYPE		layer_S	IMPLE_	MODEL	
	LAYER_SHELL	-	layer_S	IMPLE_	MODEL_s	shell
			SIMPLE_	MODEL	L 1 2 1	
END_	LAYER		STHELL	HUDEL_		

#### COLU	######################################	*######################################	Define	Column	Shells	\$ ####################################
	ТҮРЕ	SIMPLE_MODEL_	SHELL			
	HEIGHT	800				
	LOCATION	Θ	800			
END_	COLUMN_SHELL					
#### COLU	######################################	*#############	Fill Co	lumns #	#######	*****
	TYPE	SIMPLE_MODEL_	COLUMN			
	COLUMN_SHELL	SIMPLE_MODEL_	SHELL			
END_	LAYER_TYPE COLUMN	layer_SIMPLE_	MODEL			
####	######################################	+######################################	Define	Layer S	Shells	#######################################
LATE		laver SIMPLE	MODEL 4	shell		
	LOWER	0				
	UPPER	400				
END_	LAYER_SHELL					
#### LAYE	######################################	*######################################	Fill La	ayers ##	#######	******
	TYPE		layer_9	SIMPLE_M	10DEL	
	LAYER_SHELL	-	layer_S	SIMPLE_N	10DEL_s	shell
			STMPLE	MODEL 2	L 1	0
END	LAYER		SINCL_			

Anatomy

######################################	Define Column Shells ###################################
TYPE SIMPLE_MODEL_ WIDTH 300 HEIGHT 800	SHELL
LOCATION 0 END_COLUMN_SHELL	800
######################################	Fill Columns ####################################
TYPE SIMPLE MODEL	COLUMN
LAYER TYPE Lawor STMPLE	
END_COLUMN	
######################################	Define Layer Shells ###################################
TYPE layer_SIMPLE_ LOWER 0	_MODEL_shell
UPPER 400 END_LAYER_SHELL	
######################################	Fill Layers ####################################
TYPE LAYER_SHELL CELL_TYPE CELL_TYPE	layer_SIMPLE_MODEL layer_SIMPLE_MODEL_shell SIMPLE_MODEL_1 10 SIMPLE_MODEL_2 10
END LAYER	

Anatomy

####	######################################	*######################################	Define Column Shells ###################################					
COLU		SIMPLE_MODEL_	SHELL					
	HEIGHT	800						
	LOCATION	0	800					
END	_COLUMN_SHELL							
#### COLI	######################################	*######################################	Fill Columns ####################################					
	ТҮРЕ	SIMPLE MODEL	COLUMN					
	COLUMN SHELL	SIMPLE MODEL	SHELL					
	LAYER_TYPE	layer_SIMPLE_	_MODEL					
END	COLUMN							
#### Laye	######################################	*######################################	Define Layer Shells ###################################					
_/	TYPE	layer SIMPLE	MODEL shell					
	LOWER	0						
	UPPER	400						
END_	LAYER_SHELL							
#### LAYE	######################################	*###############	Fill Layers ####################################					
	TYPE		layer SIMPLE MODEL					
	LAYER_SHELL	_	layer SIMPLE MODEL shell					
	CELL_TYPE		SIMPLE_MODEL_1 10					
	CELL_TYPE		SIMPLE_MODEL_2 10					
END LAYER								

Anatomy

######################################	*######################################	Define Column Shells ###################################
TYPE S	SIMPLE_MODEL_	_SHELL
WIDTH 3 HEIGHT 8	300	
LOCATION	9	800
END_COLUMN_SHELL		
######################################	*######################################	Fill Columns ####################################
TYPE S	SIMPLE_MODEL_	_COLUMN
COLUMN SHELL S	SIMPLE MODEL	SHELL
LAYER_TYPE 1	layer_SIMPLE_	MODEL
END_COLUMN		
######################################	+######################################	Define Layer Shells ###################################
ΤΫΡΕ ι	layer_SIMPLE_	_MODEL_shell
LOWER 6)	
UPPER 4	100	
END_LAYER_SHELL		
######################################	*############	Fill Layers ####################################
TYPE		layer_SIMPLE_MODEL
LAYER_SHELL		layer_SIMPLE_MODEL_shell
CELL_TYPE		SIMPLE_MODEL_1 10
CELL_TYPE		SIMPLE_MODEL_2 10
END LAYER		

######################################	*######################################	efine Column She	lls
TYPE WIDTH	SIMPLE_MODEL 300	HELL	
HEIGHT	800		
LOCATION	Θ	800	
END_COLUMN_SHELL			
######################################	*****	ill Columns ####	******
TYPE	SIMPLE_MODEL	OLUMN	
COLUMN_SHELL	SIMPLE_MODEL	HELL	
LAYER_TYPE END_COLUMN	layer_SIMPLE	ODEL	
######################################	****	efine Layer Shel	ls ####################################
TYPE	layer SIMPLE	ODEL shell	
LOWER	0	—	
UPPER	400		
END_LAYER_SHELL			
######################################	#######################################	ill Layers #####	#######################################
TYPE		ayer_SIMPLE_MODE	L
LAYER_SHE	LL	ayer_SIMPLE_MODE	L_shell
CELL_TYPE		IMPLE_MODEL_1	10
		IMPLE_MODEL_2	10
END LATER			

#### COLI	######################################	*######################################	Define	Column	Shell	s ####################################
	TYPE	SIMPLE_MODEL_	SHELL			
	WIDTH	300				
	LOCATION	0	800			
END_	_COLUMN_SHELL					
#### COLI	######################################	*######################################	Fill Co	olumns	######	***********
	TYPE	SIMPLE_MODEL_	COLUMN			
	COLUMN_SHELL	SIMPLE_MODEL_	SHELL			
END_	_COLUMN	layer_SIMPLE_	MODEL			
####	*################	*######################################	Define	Layer	Shells	#######################################
LAY	ER_SHELL					
	TYPE	layer SIMPLE	MODEL s	shell		
	LOWER	0				
		400				
END_	LATER_SHELL					
#### Laye	######################################	*######################################	Fill La	ayers #	######	**********
	TYPE		layer_S	SIMPLE_	MODEL	
	LAYER_SHELI	_	layer_S	SIMPLE_	MODEL_	shell
	CELL_TYPE		SIMPLE_	MODEL_	1	10
			SIMPLE	MODEL_	2	10
END	LATER					

######################################	Define Column Shells ###################################
TYPE SIMPLE_MODEL_	SHELL
HEIGHT 800	
LOCATION 0	800
END_COLUMN_SHELL	
######################################	Fill Columns ####################################
TYPE SIMPLE_MODEL_	COLUMN
COLUMN_SHELL SIMPLE_MODEL_	SHELL
END_COLUMN	MODEL
######################################	Define Layer Shells
TYPE laver SIMPLE	MODEL_shell
LOWER 0	
UPPER 400	
END_LATER_SHELL	
######################################	Fill Layers ####################################
TYPE	layer_SIMPLE_MODEL
LAYER_SHELL	layer_SIMPLE_MODEL_shell
	STULFTE NODEL Z IO

Anatomy

######################################	Define Column Shells
TYPE SIMPLE_MODEL	SHELL
HEIGHT 800	
LOCATION 0	800
END_COLUMN_SHELL	
######################################	Fill Columns ####################################
TYPE SIMPLE_MODEL_	COLUMN
COLUMN_SHELL SIMPLE_MODEL	
END_COLUMN	MODEL
######################################	Define Layer Shells ###################################
TYPE layer_SIMPLE	_MODEL_shell
LOWER 0	
END_LAYER_SHELL	
######################################	Fill Layers ####################################
ТҮРЕ	layer_SIMPLE_MODEL
	layer_SIMPLE_MODEL_shell
	SIMPLE_MUDEL_I IU SIMPLE_MODEL_2 10
END LAYER	

######################################	*######################################	Define	Column	Shells	5 ####################################
TYPE WIDTH	SIMPLE_MODEL_ 300	SHELL			
HEIGHT	800				
LOCATION	0	800			
END_COLUMN_SHELL					
######################################	*######################################	Fill Co	olumns a	#######	******
TYPE	SIMPLE_MODEL_	COLUMN			
COLUMN_SHELL	SIMPLE_MODEL	SHELL			
LAYER_TYPE	layer_SIMPLE_	MODEL			
#####################	#######################################	Define	Layer	Shells	#############################
LAYER_SHELL					
TYPE	layer_SIMPLE_	MODEL_s	shell		
LOWER	0				
END LAYER SHELL	400				
	_				
######################################	*######################################	Fill La	ayers #	#######	*****
TYPE		laver S	SIMPLE	MODEL	
LAYER SHELI	_	layer S	SIMPLE	MODEL s	shell
CELL_TYPE		SIMPLE	MODEL	1 1	LO
CELL_TYPE		SIMPLE_	MODEL_	2 1	LO
END_LAYER					

##### COLUM	<i>!####################################</i>	*######################################	Define	Column	Shells	#######################################
T		SIMPLE_MODEL_	SHELL			
H	EIGHT	800				
Ĺ	OCATION	0	800			
END_C	OLUMN_SHELL					
##### COLUM	<i>!###############</i> 1N	*######################################	Fill Co	olumns a	#######	#######################################
Т	YPE	SIMPLE_MODEL_	COLUMN			
C	OLUMN_SHELL	SIMPLE_MODEL_	SHELL			
	AYER_TYPE	layer_SIMPLE_	MODEL			
END_C	.OLUMN					
#####	*###############	+######################################	Define	Laver S	Shells	#######################################
LAYER	SHELL					
Т	YPE	<pre>layer_SIMPLE_</pre>	MODEL_s	shell		
L	OWER	0				
		400				
END_L	ATER_SHELL					
##### LAYER	!#####################################	t#####################################	Fill La	ayers #;	#######	#######################################
	TYPE		layer S	SIMPLE N	MODEL	
	LAYER_SHELL	-	layer_S	SIMPLE_	MODEL_s	hell
	CELL_TYPE		SIMPLE	MODEL_	1 1	0
	CELL_TYPE		SIMPLE_	_MODEL_2	2 1	Θ
END_L	ATEK					

######################################	*######################################	Define	Column	Shells	s ####################################
TYPE WIDTH	SIMPLE_MODEL_ 300	SHELL			
HEIGHT	800	000			
END_COLUMN_SHELL	0	800			
######################################	*######################################	Fill Co	olumns	######	******
TYPE	SIMPLE_MODEL_	COLUMN			
COLUMN_SHELL	SIMPLE_MODEL_	SHELL			
LAYER_TYPE END_COLUMN	layer_SIMPLE_	MODEL			
######################################	*######################################	Define	Layer	Shells	#######################################
	laver SIMPLE	MODEL 4	shell		
LOWER	0				
UPPER	400				
END_LAYER_SHELL					
######################################	*##############	Fill La	ayers #	######	*######################################
TYPE		laver S	SIMPLE	MODEL	
LAYER SHELL	-	layer S	SIMPLE	MODEL S	shell
CELL_TYPE		SIMPLE	MODEL_	1 1	10
		SIMPLE	MODEL_	Ζ.	

######################################	*######################################	Define	Column	Shells	#######################################
	SIMPLE_MODEL_	SHELL			
HEIGHT	800				
LOCATION	Θ	800			
END_COLUMN_SHELL					
######################################	*######################################	Fill Co	olumns a	#######	*********************
TYPE	SIMPLE_MODEL_	COLUMN			
COLUMN_SHELL	SIMPLE_MODEL_	SHELL			
	layer_SIMPLE_	MODEL			
#######################################	#######################################	Define	Layer S	Shells #	#######################################
LAYER SHELL					
TYPE	layer_SIMPLE_	MODEL_s	shell		
LOWER	0				
	400				
ENU_LATER_SHELL					
######################################	*######################################	Fill La	ayers #	#######	""""""""""""""""
TYPE		layer_9	SIMPLE_	MODEL	
LAYER SHEL	-	laver_9	SIMPLE_	MODEL_s	hell
		SIMPLE	MODEL		9
		SIMPLE	MODEL	2 10	

############	########	+#############	Define	Cells	#####	####	################	#########
CELL TYPE COMPARTM	ENT	SIMPLE_MODEL soma_SIMPLE_	_1 MODEL	somaE	Ø	Θ	0	
END_CELL								
CELL TYPE COMPARTM END_CELL	ENT	SIMPLE_MODEL soma_SIMPLE_	2 MODEL	somaE	Θ	0	Θ	
############# ## Define Co ##############	######## mpartmer #########	##### nts ######						
COMPARTMENT TYPE SPIKESHA TAU_MEMB R_MEMBRA THRESHOL VMREST END_COMPARTM	PE RANE NE D ENT	soma_SIMPL spikeshape 0.020 200 -40 -60	.E_MODEI e_1k_de1 0.0 0 0 0	- fault				
############	#######	####### Defi	ine Spil	keshape	#####	####	#################	####
SPIKESHAPE TYPE VOLTAGES END_SPIKESHA	spił -38 PE	xeshape_1k_de 30 -43 -60	efault					

####	###############	############# Define	e Cells	#####	####	+++++++++++++++++++++++++++++++++++++++
CELL						
-	TYPE	SIMPLE_MODEL_1				
(COMPARTMENT	soma_SIMPLE_MODEL	somaE	Θ	Θ	Θ
END_0	CELL					
CELL						
CELL_	TVDE	STMPLE MODEL 2				
		SOMA STMPLE MODEL	somaF	ο	Θ	Θ
			Solice	U	0	0
####	################	########				
## De	efine Compartm	nents				
####	################	########				
COMP	DTMENT					
COMP			- 1			
		soma_SIMPLE_MOD	EL ofault			
-	TAIL MEMBRANE		erautt			
l i	R	200 0				
	THRESHOLD	-40 0				
١	VMREST	-60 0				
END_0	COMPARTMENT					
####	################	######### Define Sp:	ikeshape	e#####	####	#######################################
CDTW						
SPIK	ESHAPE	ikoshano 1k dofault				
	FE SP LTAGES - 3	38 30 - 43 - 60				
END 9		50 50 -45 -00				

####	############	#############	Define	Cells	#####	####	#######################################
CELL							
l	TYPE	SIMPLE MODEL	1	-	0	•	0
	COMPARIMENT	SOMA_SIMPLE_N	IODEL	somaE	Θ	Θ	0
	CELL						
CELL							
	TYPE	SIMPLE_MODEL	2				
	COMPARTMENT	soma_SIMPLE_N	10DEL	somaE	Θ	0	0
END_	CELL						
####	##############	##########					
#### ## D	efine Compar	########## tments					
####	#######################################	##########					
COMP	ARTMENT						
	TYPE	soma_SIMPLE	E_MODEL				
	SPIKESHAPE	spikeshape_	_1k_de1	rault			
		200	0.0				
	THRESHOLD	- 40	õ				
	VMREST	- 60	õ				
END_	COMPARTMENT						
####	##############	########### Defir	ne Spik	keshape	#####	####	#######################################
CDTK	ECHADE						
TY	PE	spikeshape 1k det	fault				
vo	LTAGES	-38 30 -43 -60					
END	SPIKESHAPE						

####	##############	+######################################	Define	Cells	#####	####	####	#######################################	####
CELI	_								
	TYPE	SIMPLE MODE	L 1						
	COMPARTMENT	soma SIMPLE	MODEL	somaE	0	0	0		
END	CELL								
-	-								
CELI	_								
	TYPE	SIMPLE MODE	L 2						
	COMPARTMENT	soma SIMPLE	MODEL	somaE	Θ	0	0		
END	CELL	_	_						
####	#######################################	+#########							
## [Define Compar	rtments							
####	#######################################	+##########							
COM	PARTMENT								
	TYPE	soma_SIMP	LE_MODEI						
	SPIKESHAPE	spikeshap	e_1k_de	fault					
	TAU_MEMBRANE	0.020	0.0						
	R_MEMBRANE	200	0						
	THRESHOLD	- 40	0						
	VMREST	- 60	0						
END	COMPARTMENT								
####	#######################################	############# Def	ine Spil	keshape	2#####	####	####	<i>####################################</i>	:
CDT									
2611	(ESHAPE	and backson a 11 d	- f] +						
		spikesnape_ik_d	erault						
		-38 30 -43 -60							
END	SPIKESHAPE								

####	+++++++++++++++++++++++++++++++++++++++	+######################################	Define	Cells	#####	####	#######################################
CELL							
	TYPE	SIMPLE_MODEL	_1	-			
	COMPARTMENT	soma_SIMPLE_	MODEL	somaE	Θ	0	Θ
END_	CELL						
CELL							
	TYPE	SIMPLE MODEL	. 2				
	COMPARTMENT	soma_SIMPLE_	MODEL	somaE	Θ	0	Θ
END_	CELL						
#### ## F	######################################	########					
#####	######################################	#######					
COMF	ARTMENT						
	TYPE	soma_SIMPL	E_MODE	L			
	SPIKESHAPE	spikeshape	_1k_de	fault			
		0.020	0.0				
		- 40	0				
	VMREST	- 60	õ				
END	COMPARTMENT		•				
_							
####	+######################################	######### Defi	ne Spil	keshape	*#####	####;	#######################################
CDT							
SPIK		ikashana 1k da	foult				
	PE SP	1Kesnape_ik_ue	autt				
END	SPIKESHAPE	0 50 -45 -00					

####	+###############	############## Defi	ne Cells	#####	#####	##########	#######################################
CELL	_						
	TYPE	SIMPLE_MODEL_1					
	COMPARTMENT	soma_SIMPLE_MODE	L somaE	Θ	0	Θ	
END_	CELL						
CELL							
CELL	- TYDE	STMPLE MODEL 2					
	COMPARTMENT	SOMA STMPLE MODE	l somaF	Θ	0	0	
END	CELL		E Solide	Ŭ	Ŭ	0	
####	+######################################	########					
## C)efine Compart	ments					
####	+######################################	########					
COM							
COMP			DEL				
		soma SIMPLE MU	dofault				
	TAU MEMBRANE						
		200 0	0				
	THRESHOLD	-40 0					
	VMREST	- 60 0					
END	COMPARTMENT						
-	-						
####	+################	######### Define S	pikeshap	e####	#####	###########	###########
SPIK	(ESHAPE						
	(PE s	pikeshape_1k_defaul	t				
	SPIKESHAPE	38 30 -43 -60					
	SPIKESHAPE						

#####	#######################################	############### D	efine Ce	ells	#####	####	###########################
CELL	_						
	TYPE	SIMPLE_MODEL_	1				
	COMPARTMENT	soma_SIMPLE_M	ODEL so	omaE	0	0	Θ
END_{-}	CELL						
CELL	-		2				
				maE	0	0	0
		SOMA_SIMPLE_M	UDEL SO	mae	0	0	0
####	****	########					
## C	Define Compart	ments					
####	#######################################	#########					
COMF	PARTMENT						
	TYPE	soma SIMPLE	MODEL				
	SPIKESHAPE	spikeshape	1k defau	ılt			
	TAU_MEMBRANE	0.020	0.0				
	D MEMBDANE	200	~				
	K_PIEPIDKANE	200	Θ				
	THRESHOLD	- 40	0				
	THRESHOLD VMREST	- 40 - 60	0 0 0				
END_	THRESHOLD VMREST COMPARTMENT	- 40 - 60	0 0 0				
END_	THRESHOLD VMREST COMPARTMENT	-40 -60	0 0	hana			
END_ ####	THRESHOLD VMREST COMPARTMENT	-40 -60 ######### Defin	0 0 0 e Spikes	shape	#####	####	#######################################
END_ ####	THRESHOLD VMREST COMPARTMENT	-40 -60 ########## Defin	0 0 0 e Spikes	shape	#####	####	#######################################
END_ #### SPIK	THRESHOLD VMREST COMPARTMENT	-40 -60 ########## Defin	0 0 e Spikes	shape	#####	####	#######################################
END_ #### SPIK TY	THRESHOLD VMREST COMPARTMENT ################# KESHAPE YPE s DLTAGES -	-40 -60 ########## Defin pikeshape_1k_def 38 30 -43 -60	0 0 e Spikes ault	shape	#####	####	########################
END_ #### SPIK TY VC END	THRESHOLD THRESHOLD VMREST COMPARTMENT ################ KESHAPE (PE s DLTAGES - SPTKESHAPE	-40 -60 ######### Defin pikeshape_1k_def 38 30 -43 -60	0 0 e Spikes ault	shape	#####	####	#####################

####	#################	############ Define	Cells	#####	####	#######################################
CELL						
	ТҮРЕ	SIMPLE_MODEL_1				
	COMPARTMENT	soma_SIMPLE_MODEL	somaE	Θ	0	Θ
END_	CELL					
CELL	TYDE					
		SIMPLE_MODEL_2	comaE	0	0	0
END	CELL	Solia_STHFEE_HODEE	SUNAL	0	0	0
	CLLL					
####	##################	######				
## D	efine Compartme	nts				
####	<i>##################</i> ##################	######				
COMP	ARTMENT					
	ТҮРЕ	soma_SIMPLE_MODE	L			
	SPIKESHAPE	spikeshape_ik_de	Tault			
		0.020 0.0				
		200 0				
	VMREST	-40 0				
END	COMPARTMENT	-00 0				
				_		
####	#######################################	######## Define Spi	keshape	e; * ####	####	#######################################
SPTK	FSHAPE					
TY	'PE soi	keshape 1k default				
VO	LTAGES - 38	30 - 43 - 60				
END	SPIKESHAPE					

####	+######################################	########### De	fine	Cells	#####	####	#######################################	##
CELL	- TVDE	STMDLE MODEL 1						
	COMPARTMENT	soma SIMPLE MO	DEL	somaE	0	0	0	
END_	CELL							
CELL								
CLLL	TYPE	SIMPLE MODEL 2						
	COMPARTMENT	soma_SIMPLE_MO	DEL	somaE	0	0	0	
END_	CELL							
####	*#################	#####						
## C)efine Compartmer	nts						
####	*##################	#####						
COMF	PARTMENT							
	TYPE	soma_SIMPLE_	MODEL	- -				
	TAU MEMBRANE	spikesnape_i 0.020	к_ает 0.0	autt				
	R_MEMBRANE	200	0					
	THRESHOLD	- 40	0					
END		-60	0					
####	+######################################	####### Define	Spi	keshape	2#####	####;	#######################################	
SPTK	(ESHADE							
T	PE spil	keshape 1k defa	ult					
VC	LTAGES - 38	30 -43 -60						
END_	SPIKESHAPE							

####	*##################	############ Defin	e Cells	#####	+###	#######################################
CELL						
	TYPE	SIMPLE_MODEL_1				
	COMPARTMENT	soma_SIMPLE_MODEL	somaE	Θ	0	Θ
END_	CELL					
CELL						
		SIMPLE_MODEL_2	compE	0	0	0
		SOMA_SIMPLE_MODEL	SomaE	0	0	0
END_	CELL					
####		######				
## C	efine Compartme	nts				
####	#######################################	######				
COMF	PARTMENT					
	TYPE	soma_SIMPLE_MOD	EL			
	SPIKESHAPE	<pre>spikeshape_1k_d</pre>	efault			
	TAU_MEMBRANE	0.020 0.0				
	R_MEMBRANE	200 0				
	THRESHOLD	-40 0				
	VMREST	-60 0				
END_	COMPARIMENT					
####		######## Dofing Cn	ikochan	~#####		****
####	•################	######## Deline Sh	rkesnap	2#####	+####	******
SPTK	ESHAPE					
T	'PE snil	keshape 1k default				
V	LTAGES - 38	30 - 43 - 60				
END	SPIKESHAPE					
-	-					

Spike shape



####	+##############	################## D	efine	Cells	#####	####	########	########	#######
CELL	-								
	TYPE	SIMPLE_MODEL_	1						
	COMPARTMENT	soma_SIMPLE_M	IODEL	somaE	Θ	0	0		
END_	CELL								
CELL									
CELL	TYPE	STMPLE MODEL	2						
	COMPARTMENT	soma SIMPLE M	ODEL	somaE	0	Θ	0		
END	CELL			o o ind L			•		
_									
####	+######################################	+########							
## [)efine Compart	tments							
####	+######################################	#########							
сом									
COM		soma STMPLE							
	SPTKESHAPE	snikeshane		- fault					
	TAU MEMBRANE	0.020	0.0	aace					
	R MEMBRANE	200	0						
	THRESHOLD	-40	0						
	VMREST	- 60	0						
END_	COMPARTMENT								
######################################									
SPT	(ESHAPE								
T\	PE s	spikeshape 1k def	ault						
vo	LTAGES -	-38 30 -43 -60							
END	SPIKESHAPE								
	-								

####	*###############	################ Define	Cells	#####	#####	"	###
CELL	-						
	TYPE	SIMPLE_MODEL_1					
	COMPARTMENT	soma_SIMPLE_MODEL	somaE	0	0	Θ	
END_	CELL						
CELL	-						
		SIMPLE_MODEL_2	c e me F	0	•	0	
		SOMA_SIMPLE_MODEL	SomaE	U	0	0	
END_							
####	***	+#######					
## [)efine Compartm	ients					
####	+######################################	+#######					
COMF	PARTMENT						
	TYPE	soma_SIMPLE_MODE	L				
	SPIKESHAPE	spikeshape_1k_de	efault				
	TAU MEMBRANE	0.020 0.0					
	R MEMBRANE	200 0					
	THRESHOLD	-40 0					
EN ID	VMREST	-60 0					
END_	COMPARTMENT						
		HHHHHHHH Dafina Chi	kachan				
######################################							
SPT	(ESHADE						
	/PF sr	vikeshane 1k default					
vr	ITAGES - 3	38 30 -43 -60					
END	SPIKESHAPE	0 00 -40 -00					

####	###############	############# Defin	e Cells	#####	#####	##########	+++++++++++++++++++++++++++++++++++++++
CELL							
	TYPE	SIMPLE_MODEL_1					
	COMPARTMENT	soma_SIMPLE_MODEL	somaE	Θ	0	Θ	
END_	CELL						
CELL							
CELL	TVPF	STMPLE MODEL 2					
	COMPARTMENT	Soma STMPLE MODEL	somaF	Θ	Θ	0	
END	CELL		JoingE	Ŭ		Ū	
####	#######################################	#######					
## D	efine Compartm	ents					
####	#######################################	#######					
COM							
COMP			EI				
		soma_SIMPLE_MOD	cL ofault				
	TAIL MEMBRANE		erautt				
	R MEMBRANE	200 0					
	THRESHOLD	- 40 0					
	VMREST	-60 0					
END	COMPARTMENT						
######################################							
CDT							
SPIK	LESHAPE	ikaabana 1k dafault					
	PE SP.	ikesnape_ik_default					
		0 30 -43 -00					
	SFINESHAFE						

####	#############	################ Define	Cells	#####	####	##########################	##
CELL							
	TYPE	SIMPLE_MODEL_1					
	COMPARTMENT	soma_SIMPLE_MODEL	somaE	Θ	0	Θ	
END_	CELL						
CELL							
	ТҮРЕ	SIMPLE_MODEL_2	-	•	•	0	
	COMPARIMENT	SOMA_SIMPLE_MODEL	somaE	Θ	Θ	Θ	
END_	CELL						
####		##########					
### D	efine Compar	**************************************					
####	#######################################	###########					
COMP	ARTMENT						
	TYPE	soma SIMPLE MODE	L				
	SPIKESHAPE	spikeshape ik de	fault				
	TAU_MEMBRANE	0.020 0.0					
	R_MEMBRANE	200 0					
	THRESHOLD	-40 0					
	VMREST	-60 0					
END_	COMPARTMENT						
######################################							
соти							
SPIK	ESHAPE	cnikachana 1k dafault					
		- 30 30 - 43 - 00					
	SFIRESHAFE						

Membrane Potential



Channel a

-			
COM	PARTMENT		
	TYPE	1PLE MODEL2	
	SPIKESHAPE	spikesha	ape channels
	TAU MEMBRANE	0.020	0.0
	R MEMBRANE	200	Θ
	THRESHOLD	-40	Θ
	VMREST	<u>-6</u> 0	Θ
	CHANNEL	a	
END	COMPARTMENT		
	—		

Channel a

CHANNEL Ka	a					
TY	(PE	а				
М	INITIAL	0.0		0.0		
H	INITIAL	1.0		0.0		
RE	EVERSAL POTENTIAL	- 80		0		
М	POWER	1				
H	POWER	1				
E	HALF MIN M	11				
E	HALF MIN H	- 56				
SL	OPE FACTOR M	18				
SL	_OPE_FACTOR_H	18				
UN	NITARY_G	0.12				
ST	FRENGTH	2.5				
V	TAU_VALUE_M	0.0002		9999		
V	TAU_VALUE_H	0.03	0.08	0.13	0.18	0.23
V	TAU VOLTAGE M	100				
V	TAU_VOLTAGE_H	-21	-1	10	21	
END_CHANNE	EL					

Channel m

COMF	PARTMENT					
	TYPE	soma SIMPLE MODEL2				
	SPIKESHAPE	spikeshape channels				
	TAU MEMBRANE	0.020	0.0			
	R MEMBRANE	200	0			
	THRESHOLD	-40	0			
	VMREST	-60	0			
	CHANNEL	m				
END	COMPARTMENT					
_	-					

Channel m

CHANNEL	Km			
	TYPE	m		
	M INITIAL	0.0		0.0
	REVERSAL POTENTIAL	- 80		Θ
	M POWER	1		
	E HALF MIN M	- 44		
	SLOPE FACTOR M	40	20	8.8
	TAU SCALE FACTOR M	0.303		
	UNITARY G	5		
	STRENGTH	0.00015		
END_CHAN	INEL			

Channel ahp

COMPARTMENT			
TYPE	soma SIMPLE MODEL2		
SPIKESHAPE	spikesha	pe_channels	
TAU_MEMBRANE	0.020	0.0	
R_MEMBRANE	200	Θ	
THRESHOLD	-40	Θ	
VMREST	-60	Θ	
CHANNEL	ahp1		
END_COMPARTMENT			
_			

Channel ahp

CHANNEL	Kahp				
	TYPE	ahp1			
	SEED	999999			
	M INITIAL	0.0	0.0		
	REVERSAL POTENTIAL	-80	Θ		
	M POWER	2			
	UNITARY G	6			
	STRENGTH	0.00015			
	CA SCALE FACTOR	0.000125			
	CA EXP FACTOR	2			
	CA_HALF_MIN	2.5			
	CA_TAU_SCALE_FACTOR	0.01			
END_CHAI	NNEL				
#######################################	#### STIMULUS	INJECTS	######################	#################	
--	------------------------------------	-------------	---	-------------------	---------
STIMULUS_INJECT TYPE S STIM TYPE r	SIMPLE_MODEL_ST realstim SIMPLE	IM MODEL			
INJECT S	SIMPLE_MODEL_CO	LUMN layer	_SIMPLE_MODEL	SIMPLE_MODEL_1	somaE 1
END_STIMULUS_INJECT					
#######################################	#########defin	e STIMULUS	#######################################	################	
STIMULUS					
TYPE	re	alstim_SIMP	LE_MODEL		
MODE	CU	RRENT			
PATTERN	PU	LSE			
DYN_RANGE	0	75			
TIMING	EX	АСТ			
AMP_START	4				
WIDTH	. 0	10			
TIME_START	0.	500			
TIME_END	0.	600			
END_STIMULUS					

######################	#### STIMUL	LUS INJE	CTS	#################	#######################################		
STIMULUS_INJECT TYPE STIM TYPE	SIMPLE_MODEL realstim SIMF	STIM	L				
INJECT	SIMPLE MODEL	COLUMN	lave	r SIMPLE MODEL	SIMPLE MODEL 1	somaE	1
END_STIMULUS_INJECT		-					
######################################	#########de1	fine STI	MULUS	#################	#######################################		
TYPE		realsti	m SIM	PLE MODEL			
MODE		CURRENT					
PATTERN		PULSE					
DYN_RANGE		0	75				
TIMING		EXACT					
AMP_START		4					
WIDTH		.010					
TIME_START		0.500					
TIME_END		0.600					
END_STIMULUS							

#######################################	STIMULUS INJECTS	#######################################	##################	
STIMULUS INJECT TYPE SIMPLE STIM TYPE realsti	MODEL STIM m SIMPLE MODEL			
INJECT SIMPLE_	MODEL_COLUMN laye	er_SIMPLE_MODEL	SIMPLE_MODEL_1	somaE 1
END_STIMULUS_INJECT				
#######################################	###define STIMULUS	5 ################	#######################################	
STIMULUS				
TYPE	realstim_SIN	1PLE_MODEL		
MODE	CURRENT			
PATTERN	PULSE			
DYN_RANGE	0 75			
TIMING	EXACT			
AMP_START	4			
WIDTH	.010			
TIME_START	0.500			
TIME_END	0.600			
END_STIMULUS				

#######################################	STIMULUS INJECTS	###################	#######################################	
STIMULUS_INJECT TYPE SIMPLE STIM TYPE realst INJECT SIMPLE END STIMULUS INJECT	MODEL STIM im SIMPLE MODEL MODEL_COLUMN laye	er_SIMPLE_MODEL	SIMPLE_MODEL_1	somaE 1
#######################################	####define STIMULUS	5 #################	#################	
STIMULUS				
TYPE	realstim SIN	IPLE MODEL		
MODE		—		
PATTERN	PULSE			
DYN RANGE	0 75			
TIMING	EXACT			
AMP START	4			
WIDTH	.010			
TIME START	0.500			
TIME END	0.600			
END_STIMULUS				
—				

#######################################	STIMULUS INJE	CTS ##############	#######################################	
STIMULUS_INJECT TYPE SIMPLE				
TNIECT STMDLE	IN SIMPLE MODEL	L		1 compE 1
	MODEL COLOMN	tayer SIMPLE MODEL	SIMPLE MODEL	I SUMAE I
END_STINOLOS_INSECT				
	####dofino STI	MIII IIS ##############		
,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,			****	
STIMULUS				
TYPE	realsti	m SIMPLE MODEL		
MODE	CURRENT			
PATTERN	PULSE			
DYN RANGE	Θ	75		
TIMING	EXACT			
AMP START	4			
WIDTH	.010			
TIME START	0.500			
TIME_END	0.600			
END STIMULUS				
_				

#######	###############	### STIMULUS	INJECT	S ##########	#######################################	÷	
STIMULUS TYPI STI	S_INJECT E SI M_TYPE re	MPLE_MODEL_ST alstim_SIMPLE	IM _MODEL				
INJECT SIMPLE_MODEL_COLUMN layer_SIMPLE_MODEL SIMPLE_MODEL_1 so END_STIMULUS_INJECT					1 soma	E 1	
#######	#######################################	########defin	e STIMU	ILUS ##########	#######################################	ł	
STIMULU	S						
[TYPE	rea	alstim	SIMPLE MODEL			
	MODE	CU	RRENT				
	PATTERN	PU	LSE				
	DYN RANGE	Θ	7	5			
	TIMĪNG	EX	АСТ				
	AMP START	4					
	WIDTH	. 0	10				
	TIME START	0.	500				
	TIME_END	0.	500				
END_STI	MULUS						

########################	##### STIMULU	S INJECTS	#######################################	+######################################	
STIMULUS_INJECT TYPE	SIMPLE_MODEL_S				
INJECT END_STIMULUS_INJECT	SIMPLE_MODEL_C	OLUMN la	yer_SIMPLE_MODEL	SIMPLE_MODEL_1	somaE 1
#######################################	##########defi	ne STIMUL	US ####################################	+######################################	
STIMULUS					
TYPE	r	<u>ealstim</u> S	IMPLE MODEL		
MODE	C	URRENT	—		
PATTERN	Р	ULSE			
DYN RANGE	Θ	75			
TIMING	E	ХАСТ			
AMP START	4				
WIDTH		010			
TIME START	Θ	.500			
TIME_END	Θ	.600			
END_STIMULUS					

#######################################	STIMULUS INJE	CTS ##############	#######################################	
STIMULUS_INJECT				
TYPE SIMPL	E_MODEL_STIM			
STIM_TYPE reals	tim_SIMPLE_MODE	:L		
INJECT SIMPL	E_MODEL_COLUMN	layer_SIMPLE_MODEL	. SIMPLE_MODEL_1 s	somaE 1
END_STIMULUS_INJECT				
######################################	#####define STI	MULUS ###################################	#######################################	
TYPE	realsti	m SIMPLE MODEL		
MODE	CURRENT			
PATTERN	PULSE			
DYN RANGE	0	75		
TIMING	EXACT			
AMP START	4			
WIDTH	.010			
TIME START	0.500			
TIME_END	0.600			
END_STIMULUS				

#######################################	# STIMULUS INJ	ECTS ##########	#######################################	
STIMULUS_INJECT				
TYPE SIM	PLE_MODEL_STIM			
STIM_TYPE rea	lstim_SIMPLE_MOD	EL		
INJECT SIM	PLE_MODEL_COLUMN	l layer_SIMPLE_MOU	DEL SIMPLE_MODEL_1	somaE 1
END STIMULUS INJECT				
######################################	######define ST	IMULUS #################	*****	
TYDE	roalst	im STMDLE MODEL		
MODE				
PATTERN	PULSE	75		
DYN RANGE	U	/5		
IIMING	EXACT			
AMP_START	4			
WIDTH	.010			
TIME_START	0.500			
TIME_END	0.600			
END_STIMULUS				
—				

STIMULUS INJECTS	#######################################	##################	
_MODEL_STIM im_SIMPLE_MODEL			
_MODEL_COLUMN lay	er_SIMPLE_MODEL	SIMPLE_MODEL_1	somaE 1
####define STIMULU	S ####################################	###################	
realstim_SI	MPLE_MODEL		
CURRENT			
PULSE			
<u>0 7</u> 5			
EXACT			
4			
.010			
0.500			
0.600			
	STIMULUS INJECTS MODEL_STIM im_SIMPLE_MODEL MODEL_COLUMN lay ####define STIMULU realstim_SI CURRENT PULSE 0 75 EXACT 4 .010 0.500 0.600	STIMULUS INJECTS ####################################	STIMULUS INJECTS ####################################

#######################################	### STIMULUS	INJECTS	#################	#######################################	
STIMULUS_INJECT TYPE SI STIM_TYPE re	IMPLE_MODEL_STI ealstim_SIMPLE_	MODEL			
INJECT SI END_STIMULUS_INJECT	IMPLE_MODEL_COL	UMN layer	SIMPLE_MODEL	SIMPLE_MODEL_1	somaE 1
#######################################	########define	STIMULUS	#######################################	#######################################	
STIMULUS					
TYPE	rea	lstim SIMP	LE MODEL		
MODE	CUR	RENT	—		
PATTERN	PUL	SE			
DYN RANGE	0	75			
TIMING	EXA	СТ			
AMP START	4				
WIDTH	.01	Θ			
TIME START	0.5	00			
TIME_END	0.6	00			
END_STIMULUS					

Amplitude



#####################	#####	STIMULUS IN	JECTS	##################	#######################################		
STIMULUS_INJECT TYPE	SIMPLE	MODEL_STIM					
STIM_TYPE	realst:	LM_SIMPLE_MO	DEL			-	-
INJECT	_SIMPLE_	_MODEL_COLUM	N Lay	er_SIMPLE_MODEL	SIMPLE_MODEL_1	somaE	1
END_STIMULUS_INJEC	Γ						
#######################################	#######	####define S	TIMULU	S #################	#######################################		
STIMULUS							
TYPE		reals	tim SI	MPLE MODEL			
MODE		CURRE	NT T	—			
PATTERN		PULSE					
DYN RANGE		Θ	75				
TIMING		EXACT					
AMP START		4					
WIDTH		.010	1				
TIME START		0.500					
TIME_END		0.600					
END_STIMULUS							

#######################################	STIMULUS INJE	ECTS ###############	#######################################	
STIMULUS_INJECT TYPE SIM STIM_TYPE rea	LE_MODEL_STIM	FI		
INJECT SIM	LE MODEL COLUMN	laver SIMPLE MODEL	SIMPLE MODEL 1	somaE 1
END_STIMULUS_INJECT		,		
######################################	######define ST]	IMULUS ##############	#######################################	
STIMULUS				
TYPE	realsti	im_SIMPLE_MODEL		
MODE	CURREN	T		
PATTERN	PULSE			
DYN_RANGE	Θ	75		
TIMING	EXACT			
AMP_START	4			
WIDTH	.010			
TIME START	0.500			
TIME_END	0.600			
END_STIMULUS				
_				

#######################################	STIMULUS INJE	ECTS #####	##############	################		
STIMULUS_INJECT TYPE SIMP STIM TYPE real	LE_MODEL_STIM stim SIMPLE MODE	EL				
INJECT SIMP END_STIMULUS_INJECT	LE_MODEL_COLUMN	layer_SIMP	LE_MODEL	SIMPLE_MODEL_1	somaE 1	1
#######################################	######define STI	[MULUS #####	#############	##################		
STIMULUS						
TYPE	realsti	im_SIMPLE_MC	DEL			
MODE	CURRENT	г				
PATTERN	PULSE					
DYN_RANGE	Θ	75				
TIMING	EXACT					
AMP_START	4					
WIDTH	.010					
TIME START	0.500					
TIME END	0.600					
END_STIMULUS						

#######################################	STIMULUS INJECTS	#######################################	######	
STIMULUS_INJECT TYPE TWO_CEL STIM_TYPE realsti INJECT TWO_CEL END_STIMULUS_INJECT	L_MODEL_STIM m_TWO_CELL_MODEL L_MODEL_COLUMN	layer_TWO_CELL_MODEL	TWO_CELL_MODEL_1	somaE 1
#######################################	###define STIMULUS	#######################################	######	
STIMULUS				
TYPE	realstim_TWO	_CELL_MODEL		
MODE	CURRENT			
PATTERN	PULSE			
DYN_RANGE	0 75			
TIMING	EXACT			
AMP_START	4			
WIDTH	.010			
TIME START	0.5			
TIME_END	0.6			
FREQ_START	50			
END_STIMULUS				

Width / Frequency



#########################	****
<pre># connections</pre>	
##########################	****
CONNECT	
	TWO_CELL_MODEL_COLUMN layer_TWO_CELL_MODEL TWO_CELL_MODEL_1 somaE
	TWO_CELL_MODEL_COLUMN layer_TWO_CELL_MODEL TWO_CELL_MODEL_2 somaE
	synEE_TWO_CELL_MODEL 1 0

####### <u>###########</u> ###	*######################################
<pre># connections</pre>	
##########################	*######################################
CONNECT	
	<pre>TWO_CELL_MODEL_COLUMN layer_TWO_CELL_MODEL TWO_CELL_MODEL_1 somaE</pre>
	TWO_CELL_MODEL_COLUMN layer_TWO_CELL_MODEL TWO_CELL_MODEL_2 somaE
	synEE_TWO_CELL_MODEL 1 0

########################	#########	#######	########	#######	####	+#####	+######	#####	#####	######		
<pre># connections</pre>												
#######################################	#########	#######	+########	#######	####	+#####	+######	####	+#####	######		
CONNECT												
	TWO CELL	MODEL	COLUMN	layer	TWO	CELL	MODEL	TW0	CELL	MODEL	1	somaE
	TWO CELL	MODEL	COLUMN	layer	TWO	CELL	MODEL	TW0	CELL	MODEL	2	somaE
	synEE TW	O CELL	MODEL	1 0	_		-	_			-	
			-									

#######################################	#####	#####	######	#######	######	####	#####	######	#####	#####	######		
<pre># connections</pre>													
#######################################	#####	#####	######	#######	######	####	#####	######	#####	#####	######		
CONNECT													
	TWO	CELL	MODEL	COLUMN	layer	TW0	CELL	MODEL	TW0	CELL	MODEL	1	somaE
	TWO	CELL	MODEL	COLUMN	layer	TW0	CELL	MODEL	TW0	CELL	MODEL	2	somaE
	syn	EE TWO) CELL	MODEL	1 (9							
		_		_									

########################	#################	##########	###########	+############	#########	#####	
<pre># connections</pre>							
#######################################	################	###########	###########	<i>+####################################</i>	###########	<i>\#####</i>	
CONNECT							
	TWO_CELL_MOD	EL_COLUMN	layer_TWO_	CELL_MODEL	TW0_CELL_	MODEL_1	somaE
	TWO CELL MOD	EL COLUMN	<u>layer TWO</u>	CELL_MODEL	TW0_CELL_	MODEL_2	somaE
	synEE TWO CE	<u>_L MODEL</u>	1 0				
	synEE TWO CE	L MODEL	1 0				

Probability of Connections



Probability of Connections



###############SYNAPSES SIMPLE_M	MODEL_MODEL####################################
SYNAPSE TYPE synEE_SIMPLE_M SFD_LABEL NO_SFD LEARN_LABEL NO_STDP SYN_PSG PSGexcit MAX_CONDUCT 0.4 DELAY 0.005 0.010 SYN_REVERSAL 0 0 ABSOLUTE_USE 0.25 0.1 END_SYNAPSE	MODEL
######################################	TERM SYNAPTIC DYNAMICS ####################################
SYN_FACIL_DEPRESS TYPE SFD END_SYN_FACIL_DEPRESS	NO_SFD NONE
#######################################	LONG-TERM SYNAPTIC DYNAMICS ####################################
SYN_LEARNING TYPE LEARNING END_SYN_LEARNING	NO_STDP NONE
#######################################	SYNAPTIC CONDUCTANCE WAVEFORMS ####################################
SYN_PSG TYPE PSG_FILE END_SYN_PSG	<pre>PSGexcit ./input/EPSG_Vogels_FSV1k_TAU05.inc</pre>

#################	10DEL_MODEL <mark>####################################</mark>
SYNAPSE TYPE synEE_SIMPLE_M SFD_LABEL NO_SFD LEARN_LABEL NO_STDP SYN_PSG PSGexcit MAX_CONDUCT 0.4 DELAY 0.005 0.010 SYN_REVERSAL 0 0 ABSOLUTE_USE 0.25 0.1 END_SYNAPSE	10DEL
######################################	<pre>FERM SYNAPTIC DYNAMICS ####################################</pre>
SYN_FACIL_DEPRESS TYPE SFD END_SYN_FACIL_DEPRESS	NO_SFD NONE
#######################################	LONG-TERM SYNAPTIC DYNAMICS ####################################
SYN_LEARNING TYPE LEARNING END_SYN_LEARNING	NO_STDP NONE
#######################################	SYNAPTIC CONDUCTANCE WAVEFORMS ####################################
SYN_PSG TYPE PSG_FILE END_SYN_PSG	<pre>PSGexcit ./input/EPSG_Vogels_FSV1k_TAU05.inc</pre>

###############SYNAPSES SIMPLE_I	MODEL_MODEL####################################
SYNAPSE TYPE synEE SIMPLE	MODEL
SFD_LABEL NO_SFD LEARN_LABEL NO_STDP SYN_PSG PSGevcit	
MAX_CONDUCT 0.4 DELAY 0.005 0.010	
ABSOLUTE_USE 0.25 0.1 END_SYNAPSE	
######################################	TERM SYNAPTIC DYNAMICS ####################################
SYN_FACIL_DEPRESS TYPE SFD	NO_SFD NONE
END_SYN_FACIL_DEPRESS	
SYN LEARNING	LUNG-TERM STNAFTIC DINAMICS ####################################
TYPE LEARNING END SYN LEARNING	NO_STDP NONE
	SYNAPTIC CONDUCTANCE WAVEFORMS ####################################
SYN PSG	STRAFTIC CONDUCTANCE WAVEFORTS ####################################
TYPE PSG_FILE END_SYN_PSG	<pre>PSGexcit ./input/EPSG_Vogels_FSV1k_TAU05.inc</pre>

###############SYNAPSES SIMPLE_I	MODEL_MODEL####################################
SYNAPSE	
TYPE SVNEE SIMPLE	10DEL
SFD LABEL NO SFD	
LEARN LABEL NO STDP	
SYN PSG PSGexcit	
MAX CONDUCT 0.4	
DELAY 0.005 0.010	
SYN REVERSAL 0 0	
ABSOLUTE USE 0.25 0.1	
END SYNAPSE	
_ ####################################	TERM SYNAPTIC DYNAMICS
SYN FACIL DEPRESS	
TYPE	NO SFD
SFD	NONE
END SYN FACIL DEPRESS	
#######################################	LONG-TERM SYNAPTIC DYNAMICS ####################################
SYN LEARNING	
TYPE	NO STDP
LEARNING	NONE
END SYN LEARNING	
#######################################	SYNAPTIC CONDUCTANCE WAVEFORMS ####################################
SYN_PSG	
ТҮРЕ	PSGexcit
PSG FILE	./input/EPSG Vogels FSV1k TAU05.inc
END_SYN_PSG	

###############SYNAPSE	S SIMPLE_	10DEL_MODEL####################################
SYNAPSE		
TYPE synE	E SIMPLE N	10DEL
SFD LABEL	NO SFD	_
LEARN_LABEL	NO_STDP	
SYN_PSG PSGe	xcit	
MAX_CONDUCT 0.4		
DELAY 0.00	5 0.010	
SYN_REVERSAL 0	Θ	
ABSOLUTE_USE 0.25	0.1	
END_SYNAPSE		
######################################	## SHORT- ⁻	FERM SYNAPTIC DYNAMICS
		NO SED
SED		NONE
END SYN FACTL DEPRESS		NONE
#######################################	##########	LONG-TERM SYNAPTIC DYNAMICS ####################################
SYN LEARNING		
TYPE		NO STDP
LEARNING		NONE
END SYN LEARNING		
#######################################	#########	SYNAPTIC CONDUCTANCE WAVEFORMS ####################################
SYN_PSG		
TYPE		PSGexcit
PSG_FILE		./input/EPSG_Vogels_FSV1k_TAU05.inc
END_SYN_PSG		

################SYNAPSES SIMPLE_MODEL_MODEL####################################				
SYNAPSEsynEE_SIMPLE_ISFD_LABELNO_SFDLEARN_LABELNO_STDPSYN_PSGPSGexcitMAX_CONDUCT0.4DELAY0.005SYN_REVERSAL0ABSOLUTE_USE0.25O.1END_SYNAPSE	MODEL			
######################################	TERM SYNAPTIC DYNAMICS ####################################			
SYN_FACIL_DEPRESS TYPE SFD END_SYN_FACIL_DEPRESS	NO_SFD NONE			
#######################################	LONG-TERM SYNAPTIC DYNAMICS ####################################			
SYN_LEARNING TYPE LEARNING END_SYN_LEARNING	NO_STDP NONE			
#######################################	SYNAPTIC CONDUCTANCE WAVEFORMS ####################################			
SYN_PSG TYPE PSG_FILE END_SYN_PSG	PSGexcit ./input/EPSG_Vogels_FSV1k_TAU05.inc			

###############SYNAPSES SIMPLE_N	MODEL_MODEL####################################
SYNAPSEsynEE_SIMPLE_ISFD_LABELNO_SFDLEARN_LABELNO_STDPSYN_PSGPSGexcitMAX_CONDUCT0.4DELAY0.005SYN_REVERSAL0ABSOLUTE_USE0.25END_SYNAPSE	MODEL
######################################	TERM SYNAPTIC DYNAMICS ####################################
SYN_FACIL_DEPRESS TYPE SFD END_SYN_FACIL_DEPRESS	NO_SFD NONE
#######################################	LONG-TERM SYNAPTIC DYNAMICS ####################################
SYN_LEARNING TYPE LEARNING END_SYN_LEARNING	NO_STDP NONE
****	SYNAPTIC CONDUCTANCE WAVEFORMS ####################################
SYN_PSG TYPE PSG_FILE END_SYN_PSG	PSGexcit ./input/EPSG_Vogels_FSV1k_TAU05.inc

###############SYNAPSES SIMPLE_I	MODEL_MODEL####################################
SYNAPSE TYPE synEE_SIMPLE_I SFD_LABEL NO_SFD LEARN_LABEL NO_STDP SYN_PSG PSGexcit MAX_CONDUCT 0.4 DELAY 0.005 0.010 SYN_REVERSAL 0 0 ABSOLUTE_USE 0.25 0.1 END_SYNAPSE	MODEL
######################################	TERM SYNAPTIC DYNAMICS
SYN_FACIL_DEPRESS TYPE SFD END_SYN_FACIL_DEPRESS	NO_SFD NONE
#######################################	LONG-TERM SYNAPTIC DYNAMICS ####################################
SYN_LEARNING TYPE LEARNING END_SYN_LEARNING	NO_STDP NONE
#######################################	SYNAPTIC CONDUCTANCE WAVEFORMS ####################################
SYN_PSG TYPE PSG_FILE END_SYN_PSG	PSGexcit ./input/EPSG_Vogels_FSV1k_TAU05.inc

Conductance Strength



Conductance Strength



Conductance Strength



##############SYNAPSES SIMPLE	MODEL MODEL####################################
SYNAPSE TYPE synEE_SIMPLE_N SFD_LABEL NO_SFD LEARN_LABEL NO_STDP SYN_PSG PSGexcit MAX_CONDUCT 0.4 DELAY 0.005 0.010 SYN_REVERSAL 0 0 ABSOLUTE_USE 0.25 0.1 END_SYNAPSE	MODEL
######################################	TERM SYNAPTIC DYNAMICS ####################################
SYN_FACIL_DEPRESS TYPE SFD END_SYN_FACIL_DEPRESS	NO_SFD NONE
#######################################	LONG-TERM SYNAPTIC DYNAMICS ####################################
SYN_LEARNING TYPE LEARNING END_SYN_LEARNING	NO_STDP NONE
*****	SYNAPTIC CONDUCTANCE WAVEFORMS ####################################
SYN_PSG TYPE PSG_FILE END_SYN_PSG	<pre>PSGexcit ./input/EPSG_Vogels_FSV1k_TAU05.inc</pre>
Synapses

##############SYNAPSES SIMPLE	MODEL MODEL####################################
SYNAPSE TYPE synEE_SIMPLE_I SFD_LABEL NO_SFD LEARN_LABEL NO_STDP SYN_PSG PSGexcit MAX_CONDUCT 0.4 DELAY 0.005 0.010 SYN_REVERSAL 0 0 ABSOLUTE_USE 0.25 0.1 END_SYNAPSE	
######################################	IERM SYNAPIIC DYNAMICS ####################################
SYN_FACIL_DEPRESS TYPE SFD END_SYN_FACIL_DEPRESS	NO_SFD NONE
#######################################	LONG-TERM SYNAPTIC DYNAMICS ####################################
SYN_LEARNING TYPE LEARNING END_SYN_LEARNING	NO_STDP NONE
*****	SYNAPTIC CONDUCTANCE WAVEFORMS ####################################
SYN_PSG TYPE PSG_FILE END_SYN_PSG	<pre>PSGexcit ./input/EPSG_Vogels_FSV1k_TAU05.inc</pre>

Synapses

##############SYNAPSES SIMPLE_!	MODEL_MODEL####################################
SYNAPSE TYPE synEE_SIMPLE_I SFD_LABEL NO_SFD LEARN_LABEL NO_STDP SYN_PSG PSGexcit MAX_CONDUCT 0.4 DELAY 0.005 0.010 SYN REVERSAL 0 0 ABSOLUTE_USE 0.25 0.1 END_SYNAPSE	
################################# SHORI- SYN FACIL DEPRESS	IERM SYNAPIIC DYNAMICS ####################################
TŸPE SFD	NO_SFD NONE
END_SYN_FACIL_DEPRESS	
*****	LONG-TERM SYNAPTIC DYNAMICS ####################################
SYN_LEARNING TYPE LEARNING	NO_STDP NONE
END_SYN_LEARNING	
#######################################	SYNAPTIC CONDUCTANCE WAVEFORMS ####################################
SYN_PSG TYPE PSG_FILE END_SYN_PSG	<pre>PSGexcit ./input/EPSG_Vogels_FSV1k_TAU05.inc</pre>

###############SYNAPSES SIMPLE [MODEL MODEL####################################
SYNAPSE	
TYPE synEE SIMPLE	10DEL
SFD LABEL NO SFD	
LEARN LABEL NO STDP	
SYN PSG PSGexcit	
MAX CONDUCT 0.4	
DELĀY 0.005 0.010	
SYN REVERSAL 0 0	
ABSOLUTE USE 0.25 0.1	
END SYNAPSE	
######################################	TERM SYNAPTIC DYNAMICS
SYN_FACIL_DEPRESS	
ТҮРЕ	NO_SFD
SFD	NONE
END_SYN_FACIL_DEPRESS	
#######################################	LUNG-TERM SYNAPTIC DYNAMICS ####################################
	NUNE
END_STN_LEARNING	

**************************************	STNAFTIC CONDUCTANCE WAVEFURIS ####################################
SYN PSG	
TYPE	PSGercit
PSG FTLF	/input/EPSG Vogels ESV1k TAU05 inc
END SYN PSG	The state of the s

###############SYN	NAPSES SIMPLE_	MODEL_MODEL##;	##########	<i>!#########################</i>
SYNAPSE TYPE SFD_LABEL LEARN_LABEL SYN_PSG MAX_CONDUCT DELAY SYN_REVERSAL ABSOLUTE_USE END_SYNAPSE	synEE_SIMPLE_ FACILIT NO_STDP PSGexcit 0.4 0.005 0.010 0 0 0.25 0.1	MODEL		
#######################################	###### SHORT-	TERM SYNAPTIC	DYNAMICS	#######################################
SYN_FACIL_DEPRESS TYPE SFD FACIL_TAU DEPR_TAU END_SYN_FACIL_DEPF	RESS	FACILITATION BOTH 0.376 0.045	0.0 0.0	

###;	###########SYI	NAPSES	SIMPLE_	MODEL_MODEL###	#########	#######################################
SYN	APSE					
	TYPE	synEE	SIMPLE	MODEL		
	SFD LABEL		FACILIT	ATION		
	LEARN LABEL		NO STDP	,		
	SYN PSG	PSGexc	it –			
	MAX ^{CONDUCT}	0.4				
	DELAY	0.005	0.010			
	SYN REVERSAL	0 0				
	ABSOLUTE USE	0.25	0.1			
END	SYNAPSE					
###;	###############	#######	SHORT-	TERM SYNAPTIC	DYNAMICS	#######################################
SYN	FACIL DEPRESS					
	TYPE			FACILITATION		
	SFD			BOTH		
	FACIL TAU			0.376	0.0	
	DEPR TAU			0.045	0.0	
END	_SYN_FACIL_DEP	RESS				

###############SYNAPSES SIMPLE MODEL	MODEL####################################
--------------------------------------	---

	TYPE	synEE SIMPLE MODEL
	SFD LABEL	FACILITATION
	LEARN LABEL	NO STDP
	SYN PSG	PSGexcit
	MAX_CONDUCT	0.4
	DELAY	0.005 0.010
	SYN REVERSAL	0 0
	ABSOLUTE USE	0.25 0.1
END	SYNAPSE	

SYN FACI	L DEPRESS
----------	-----------

_			
	ТҮРЕ	FACILITATION	
	SFD	BOTH	
	FACIL TAU	0.376	0.0
	DEPR TAU	0.045	0.0
END_SYN_	FACIL_DEPRESS		
4			

<i>###############</i> SY	NAPSES SIMP	_E_MODEL_MODE	L################	#######################################
SYNAPSE				
TYPE	synEE SIMP	LE MODEL		
SFD LABEL	FACI	LITATION		
LEARN LABEL	NO S	ГDР		
SYN PSG	PSGexcit [_]			
MAX ^{CONDUCT}	0.4			
DELAY	0.005 0.03	10		
SYN REVERSAL	00			
ABSOLUTE USE	0.25 0.3	1		
END_SYNAPSE				
#######################################	###### SHOI	RT-TERM SYNAP	TIC DYNAMICS ##	"######################################
SYN FACIL DEPRESS				
TYPE		FACILITAT	ION	
SFD		BOTH		
FACIL TAU		0.376	0.0	
DEPR TAU		0.045	0.0	
END_SYN_FACIL_DEP	RESS			

###############SY	NAPSES SIMPLE_MODEL_MODEL####################################
SYNAPSE	
TYPE	synEE_SIMPLE_MODEL
SFD_LABEL	FACILITATION
LEARN_LABEL	NO_STDP
SYN_PSG	PSGexcit
MAX_CONDUCT	0.4
DELAY	0.005 0.010
SYN_REVERSAL	0 0
ABSOLUTE_USE	0.25 0.1
END_SYNAPSE	
#######################################	####### SHORT-TERM SYNAPTIC DYNAMICS ####################################
SYN FACIL DEPRESS	
TYPE	FACILITATION
SFD	BOTH
FACIL_TAU	0.376 0.0
DEPR_TAU	0.045 0.0
END_SYN_FACIL_DEP	RESS

###############SY	NAPSES SIMPLE_MODEL_MODEL####################################
SYNAPSE	
TYPE	synEE SIMPLE MODEL
SFD LABEL	FACILITATION
LEARN LABEL	NO STDP
SYN PSG	PSGexcit
MAX ^{CONDUCT}	0.4
DELAY	0.005 0.010
SYN_REVERSAL	ΘΘ
ABSOLUTE_USE	0.25 0.1
END_SYNAPSE	
#####################	####### SHORT-TERM SYNAPTIC DYNAMICS ####################################
SYN FACIL DEPRESS	
TYPE	FACILITATION
SFD	ВОТН
FACIL TAU	0.376 0.0
DEPR TAU	0.045 0.0
END_SYN_FACIL_DEP	RESS

###############SYNAPSES_SIMPLE_	MODEL MODEL####################################
SYNAPSE	
TYPE synEE SIMPLE	10DEL
SFD LABEL NO SFD	
LEARN LABEL NO STDP	
SYN PSG PSGexcit	
MAX ^{CONDUCT 0.4}	
DELAY 0.005 0.010	
SYN REVERSAL 0 0	
ABSOLUTE USE 0.25 0.1	
END SYNAPSE	
—	
######################################	TERM SYNAPTIC DYNAMICS
SVN FACTI DEDDESS	
SED	
END SVN FACTI DEDESS	NONE
END_STN_FACIL_DEFRESS	
<i></i>	LONG-TERM SYNAPTIC DYNAMICS ####################################
SYN LEARNING	
TYPE	NO STOP
LEARNING	NONE
END SYN LEARNING	
#######################################	SYNAPTIC CONDUCTANCE WAVEFORMS ####################################
SYN PSG	
ТҮРЕ	PSGexcit
PSG FILE	/input/EPSG Vogels FSV1k TAU05.inc
END SYN PSG	.,p.c., co_ rogeco_rorincoorine

###############SYNAPSES SIMPLE_M	10DEL_MODEL####################################
SYNAPSE TYPE synEE_SIMPLE_M SFD_LABEL NO_SFD LEARN_LABEL HEBBIAN SYN_PSG PSGexcit MAX_CONDUCT 0.4 DELAY 0.005 0.010 SYN_REVERSAL 0 0 ABSOLUTE_USE 0.25 0.1 END_SYNAPSE	10DEL
######################################	TERM SYNAPTIC DYNAMICS ####################################
SYN_FACIL_DEPRESS TYPE SFD END_SYN_FACIL_DEPRESS	NO_SFD NONE
#######################################	LONG-TERM SYNAPTIC DYNAMICS ####################################
SYN_LEARNING TYPE LEARNING LEARNING_SHAPE NEG_HEB_WINDOW POS_HEB_WINDOW POS_HEB_PEAK_DELTA_USE NEG_HEB_PEAK_DELTA_USE POS_HEB_PEAK_TIME NEG_HEB_PEAK_TIME NEG_HEB_PEAK_TIME END_SYN_LEARNING	HEBBIAN BOTH EXPONENT 0.1 0.0 0.1 0.0 0.005 0.0 0.0055 0.0 0.02 0.0 0.02 0.0

########	#######SYN	APSES	SIMPLE_	MODEL_MODE	L######	####	#########	#######	########	ŧ
SYNAPSE										
TYPE	Ξ	synEE	SIMPLE	MODEL						
SFD	LABEL	-	NO SFD							
LEAF	RN LABEL		HEBBIAN	1						
SYN	PSG	PSGexc	it							
MAX	CONDUCT	0.4								
DEL	ĀY	0.005	0.010							
SYN	REVERSAL	0 0								
ABS	DLUTE USE	0.25	0.1							
END SYNA	APSE _									
_										
########	############	#######	SHORT-	TERM SYNAP	TIC DYN	IAMI	CS ######	#######	########	<i> #####</i>
SYN_FACI	IL_DEPRESS									
	TYPE			NO_SFD						
	SFD			NONE						
END_SYN_	_FACIL_DEPR	RESS								
########	*###########		#######	LONG-TERM	SYNAPT		DYNAMICS	######	########	+####
SYN LEAF	RNING									
_	TYPE			HEBBIAN						
	LEARNING			BOTH						
	LEARNING S	SHAPE		EXPONENT						
	NEG HEB WI	INDOW		0.1		0.0				
	POS HEB WI	INDOW		0.1		0.0				
	POS HEB PE	AK DEL	TA USE	0.005		0.0				
	NEG HEB PE	EAK DEL	TAUSE	0.0055		0.0				
	POS HEB PE	EAK TIM	E	0.02		0.0				
	NEG HEB PE	EAK ^T IM	E	0.02		0.0				
END SYN	LEARNING	_								
	_									

##############SYNAPSES SIMPLE_M	ODEL_MODEL####################################
SYNAPSE TYPE synEE_SIMPLE_M SFD_LABEL NO_SFD LEARN_LABEL HEBBIAN SYN_PSG PSGexcit MAX_CONDUCT 0.4 DELAY 0.005 0.010 SYN_REVERSAL 0 0 ABSOLUTE_USE 0.25 0.1 END_SYNAPSE	IODEL
######################################	ERM SYNAPTIC DYNAMICS ####################################
SYN_FACIL_DEPRESS TYPE SFD END_SYN_FACIL_DEPRESS ###################################	NO_SFD NONE LONG-TERM SYNAPTIC DYNAMICS ####################################
SYN_LEARNING TYPE LEARNING LEARNING_SHAPE NEG_HEB_WINDOW POS_HEB_WINDOW POS_HEB_PEAK_DELTA_USE NEG_HEB_PEAK_DELTA_USE POS_HEB_PEAK_TIME NEG_HEB_PEAK_TIME NEG_HEB_PEAK_TIME END_SYN_LEARNING	HEBBIAN BOTH EXPONENT 0.1 0.0 0.1 0.0 0.005 0.0 0.0055 0.0 0.02 0.0 0.02 0.0

###############SYNAPSES SIMPLE_M	MODEL_MODEL####################################
SYNAPSE TYPE synEE_SIMPLE_I SFD_LABEL NO_SFD LEARN_LABEL HEBBIAN SYN_PSG PSGexcit MAX_CONDUCT 0.4 DELAY 0.005 0.010 SYN_REVERSAL 0 0 ABSOLUTE_USE 0.25 0.1 END_SYNAPSE	10DEL
################################ SHORT- ⁻	TERM SYNAPTIC DYNAMICS ####################################
SYN_FACIL_DEPRESS TYPE SFD END_SYN_FACIL_DEPRESS	NO_SFD NONE
#######################################	LONG-TERM SYNAPTIC DYNAMICS ####################################
SYN_LEARNING TYPE LEARNING NEG_HEB_WINDOW POS_HEB_WINDOW POS_HEB_PEAK_DELTA_USE NEG_HEB_PEAK_DELTA_USE POS_HEB_PEAK_TIME NEG_HEB_PEAK_TIME NEG_HEB_PEAK_TIME END_SYN_LEARNING	HEBBIAN BOTH EXPONENT 0.1 0.0 0.11 0.0 0.0055 0.0 0.002 0.0 0.02 0.0

###############SYNAPSES SIMPLE_M	IODEL_MODEL####################################
SYNAPSE TYPE synEE_SIMPLE_M SFD_LABEL NO_SFD LEARN_LABEL HEBBIAN SYN_PSG PSGexcit MAX_CONDUCT 0.4 DELAY 0.005 0.010 SYN_REVERSAL 0 0 ABSOLUTE_USE 0.25 0.1 END_SYNAPSE	IODEL
######################################	ERM SYNAPTIC DYNAMICS ####################################
SYN_FACIL_DEPRESS TYPE SFD END_SYN_FACIL_DEPRESS ###################################	NO_SFD NONE LONG-TERM SYNAPTIC DYNAMICS ####################################
SYN_LEARNING TYPE LEARNING LEARNING SHAPE NEG_HEB_WINDOW POS_HEB_WINDOW POS_HEB_PEAK_DELTA_USE NEG_HEB_PEAK_DELTA_USE POS_HEB_PEAK_TIME NEG_HEB_PEAK_TIME NEG_HEB_PEAK_TIME END_SYN_LEARNING	HEBBIAN BOTH EXPONENT 0.1 0.0 0.005 0.0 0.0055 0.0 0.02 0.0 0.02 0.0

##############SYNAPSES SIMPLE_M	10DEL_MODEL####################################
SYNAPSE TYPE synEE_SIMPLE_M SFD_LABEL NO_SFD LEARN_LABEL HEBBIAN SYN_PSG PSGexcit MAX_CONDUCT 0.4 DELAY 0.005 0.010 SYN_REVERSAL 0 0 ABSOLUTE_USE 0.25 0.1 END_SYNAPSE	IODEL
######################################	TERM SYNAPTIC DYNAMICS ####################################
SYN_FACIL_DEPRESS TYPE SFD END_SYN_FACIL_DEPRESS	NO_SFD NONE
#######################################	LONG-TERM SYNAPTIC DYNAMICS ####################################
SYN_LEARNING TYPE LEARNING LEARNING SHAPE NEG HEB WINDOW POS_HEB_WINDOW POS_HEB_PEAK_DELTA_USE NEG_HEB_PEAK_DELTA_USE	HEBBIAN BOTH EXPONENT 0.1 0.0 0.1 0.0 0.005 0.0 0.0055 0.0
POS_HEB_PEAK_TIME NEG_HEB_PEAK_TIME END_SYN_LEARNING	0.02 0.0 0.02 0.0

################SYNAPSES SIMPLE_MODEL_MODEL####################################	<i>!###</i>
SYNAPSE TYPE SynEE_SIMPLE_MODEL SFD_LABEL NO_SFD LEARN_LABEL HEBBIAN SYN_PSG PSGexcit MAX_CONDUCT 0.4 DELAY 0.005 0.010 SYN_REVERSAL 0 0 ABSOLUTE_USE 0.25 0.1 END_SYNAPSE	
######################################	<i>!#######</i> #
SYN_FACIL_DEPRESS TYPE NO_SFD SFD NONE END_SYN_FACIL_DEPRESS	
######################################	+#######
SYN_LEARNING HEBBIAN LEARNING BOTH LEARNING_SHAPE EXPONENT NEG_HEB_WINDOW 0.1 0.0 POS_HEB_WINDOW 0.1 0.0 POS_HEB_PEAK_DELTA_USE 0.0055 0.0 NEG_HEB_PEAK_DELTA_USE 0.002 0.0 NEG_HEB_PEAK_TIME 0.02 0.0 NEG_HEB_PEAK_TIME 0.02 0.0 NEG_HEB_PEAK_TIME 0.02 0.0	

###############SYNAPSES SIMP	LE_MODEL_MODEL####################################
SYNAPSE TYPE synEE_SIMP SFD_LABEL NO_S LEARN_LABEL HEBB SYN_PSG PSGexcit MAX_CONDUCT 0.4 DELAY 0.005 0.0 SYN_REVERSAL 0 0 ABSOLUTE_USE 0.25 0. END_SYNAPSE	LE_MODEL FD IAN 10 1
######################################	RT-TERM SYNAPTIC DYNAMICS ####################################
SYN_FACIL_DEPRESS TYPE SFD END_SYN_FACIL_DEPRESS	NO_SFD NONE
#######################################	### LONG-TERM SYNAPTIC DYNAMICS ####################################
SYN_LEARNING TYPE LEARNING LEARNING_SHAPE NEG_HEB_WINDOW POS_HEB_WINDOW POS_HEB_PEAK_DELTA_U NEG_HEB_PEAK_TIME NEG_HEB_PEAK_TIME NEG_HEB_PEAK_TIME END_SYN_LEARNING	HEBBIAN BOTH EXPONENT 0.1 0.0 0.1 0.0 SE 0.0055 0.0 SE 0.0055 0.0 0.02 0.0 0.02 0.0
END_SYNAPSE ############################### SYN_FACIL_DEPRESS END_SYN_FACIL_DEPRESS ###################################	RT-TERM SYNAPTIC DYNAMICS ####################################

##############SYNAPSES SIMPLE_M	10DEL_MODEL####################################
SYNAPSE TYPE synEE_SIMPLE_M SFD_LABEL NO_SFD LEARN_LABEL HEBBIAN SYN_PSG PSGexcit MAX_CONDUCT 0.4 DELAY 0.005 0.010 SYN_REVERSAL 0 0 ABSOLUTE_USE 0.25 0.1 END_SYNAPSE	10DEL
######################################	<pre>FERM SYNAPTIC DYNAMICS ####################################</pre>
SYN_FACIL_DEPRESS TYPE SFD END_SYN_FACIL_DEPRESS	NO_SFD NONE
#######################################	LONG-TERM SYNAPTIC DYNAMICS ####################################
SYN_LEARNING TYPE LEARNING LEARNING_SHAPE NEG_HEB_WINDOW POS_HEB_WINDOW POS_HEB_PEAK_DELTA_USE NEG_HEB_PEAK_TIME NEG_HEB_PEAK_TIME END_SYN_LEARNING	HEBBIAN BOTH EXPONENT 0.1 0.0 0.1005 0.0 0.0055 0.0 0.02 0.0

##############SYNAPSES SIMPLE_MODEL_M	ODEL####################################
SYNAPSE TYPE synEE_SIMPLE_MODEL SFD_LABEL NO_SFD LEARN_LABEL HEBBIAN SYN_PSG PSGexcit MAX_CONDUCT 0.4 DELAY 0.005 0.010 SYN_REVERSAL 0 0 ABSOLUTE_USE 0.25 0.1 END_SYNAPSE	
######################################	NAPTIC DYNAMICS ####################################
SYN_FACIL_DEPRESS TYPE NO_SFD SFD NONE END_SYN_FACIL_DEPRESS	
######################################	ERM SYNAPTIC DYNAMICS ####################################
SYN_LEARNING TYPE HEBBIA LEARNING BOTH LEARNING_SHAPE EXPONE NEG_HEB_WINDOW 0.1 POS_HEB_WINDOW 0.1 POS_HEB_PEAK_DELTA_USE 0.005 NEG_HEB_PEAK_DELTA_USE 0.002 NEG_HEB_PEAK_TIME 0.02 NEG_HEB_PEAK_TIME 0.02 NEG_HEB_PEAK_TIME 0.02	N NT 0.0 0.0 0.0 0.0 0.0 0.0
LND_STN_LEARNING	

###############SYNAPSES SIMPLE_M	10DEL_MODEL####################################
SYNAPSE TYPE synEE_SIMPLE_M SFD_LABEL NO_SFD LEARN_LABEL HEBBIAN SYN_PSG PSGexcit MAX_CONDUCT 0.4 DELAY 0.005 0.010 SYN_REVERSAL 0 0 ABSOLUTE_USE 0.25 0.1 END_SYNAPSE	10DEL
######################################	TERM SYNAPTIC DYNAMICS ####################################
SYN_FACIL_DEPRESS TYPE SFD END_SYN_FACIL_DEPRESS	NO_SFD NONE
*****	LONG-TERM SYNAPTIC DYNAMICS ####################################
SYN_LEARNING TYPE LEARNING LEARNING_SHAPE NEG_HEB_WINDOW POS_HEB_WINDOW POS_HEB_PEAK_DELTA_USE NEG_HEB_PEAK_DELTA_USE POS_HEB_PEAK_TIME NEG_HEB_PEAK_TIME	HEBBIAN BOTH EXPONENT 0.1 0.0 0.01 0.0 0.005 0.0 0.0055 0.0 0.02 0.0
END_SYN_LEARNING	

#######################################	SIMPLE_MODEL_MODEL REPORTS ####################################
REPORT	
TYPE	VOLTAGE CELL 1
CELLS	SIMPLE MODEL COLUMN layer SIMPLE MODEL SIMPLE MODEL 1 somaE
PROB	1
REPORT ON	VOLTAGE
FILENAME	SIMPLE MODEL 1 VOLTAGE E.txt
ASCII	
FREQUENCY	1
TIME_START	Θ
TIME_END	100
END_REPORT	
REPORT	
TYPE	VOLTAGE CELL 2
CELLS	SIMPLE_MODEL_COLUMN layer_SIMPLE_MODEL SIMPLE_MODEL_2 somaE
PROB	1
REPORT_ON	VOLTAGE
FILENAME	SIMPLE_MODEL_2_VOLTAGE_E.txt
ASCII	
FREQUENCY	1
TIME_START	Θ
TIME_END	100
END_REPORT	

#######################################	SIMPLE MODEL MODEL REPORTS ####################################
REPORT TYPE CELLS PROB	VOLTAGE_CELL_1 SIMPLE_MODEL_COLUMN layer_SIMPLE_MODEL SIMPLE_MODEL_1 somaE 1
FILENAME ASCII FREQUENCY TIME_START	SIMPLE_MODEL_1_VOLTAGE_E.txt
END_REPORT	100
REPORT TYPE CELLS PROB REPORT_ON FILENAME ASCII FREQUENCY TIME_START TIME_END END REPORT	VOLTAGE_CELL_2 SIMPLE_MODEL_COLUMN layer_SIMPLE_MODEL SIMPLE_MODEL_2 somaE 1 VOLTAGE SIMPLE_MODEL_2_VOLTAGE_E.txt 1 0 100

###;	#################	SIMPLE_MODEL_MODEL REPORTS ####################################
REP	ORT	
	TYPE	VOLTAGE CELL 1
	CELLS	SIMPLE_MODEL_COLUMN layer_SIMPLE_MODEL SIMPLE_MODEL_1 somaE
	PROB	1
	REPORT_ON	VOLTAGE
	FILENAME	SIMPLE_MODEL_1_VOLTAGE_E.txt
	ASCII	
	FREQUENCY	1
	TIME_START	Θ
	TIME_END	100
END	REPORT	
REP	ORT	
	TYPE	VOLTAGE CELL 2
	CELLS	SIMPLE MODEL COLUMN layer SIMPLE MODEL SIMPLE MODEL 2 somaE
	PROB	1
	REPORT ON	VOLTAGE
	FILENAME	SIMPLE_MODEL_2_VOLTAGE_E.txt
	ASCII	
	FREQUENCY	1
	TIME_START	0
	TIME_END	100
END	REPORT	

################	### SIMPLE_MODEL_MODEL REPORTS ####################################
REPORT	
TYPE	VOLTAGE CELL 1
CELLS	SIMPLE MODEL COLUMN layer SIMPLE MODEL SIMPLE MODEL 1 somaE
PROB	1
REPORT ON	VOLTAGE
FILENAME	SIMPLE MODEL 1 VOLTAGE E.txt
ASCII	
FREQUENCY	1
TIME START	Θ
TIME_END	100
END_REPORT	
REPORT	
TYPE	VOLTAGE CELL 2
CELLS	SIMPLE MODEL COLUMN layer SIMPLE MODEL SIMPLE MODEL 2 somaE
PROB	1
REPORT ON	VOLTAGE
FILENAME	SIMPLE MODEL 2 VOLTAGE E.txt
ASCII	
FREQUENCY	1
TIME START	Θ
TIME_END	100
END_REPORT	

####	#################	SIMPLE_MODEL_MODEL REPORTS ####################################
REPO	RT	
	TYPE	VOLTAGE CELL 1
	CELLS	<pre>SIMPLE_MODEL_COLUMN layer_SIMPLE_MODEL SIMPLE_MODEL_1 somaE</pre>
	PROB	1
	REPORT_ON	VOLTAGE
	FILENAME	SIMPLE_MODEL_1_VOLTAGE_E.txt
	ASCII	
	FREQUENCY	1
	TIME_START	Θ
	TIME_END	100
END_	REPORT	
REPO	RT	
	TYPE	VOLTAGE CELL 2
	CELLS	SIMPLE MODEL COLUMN layer SIMPLE MODEL SIMPLE MODEL 2 somaE
	PROB	1
	REPORT ON	VOLTAGE
	FILENAME	SIMPLE MODEL 2 VOLTAGE E.txt
	ASCII	
	FREQUENCY	1
	TIME_START	Θ
	TIME_END	100
END_	REPORT	

####	*#################	SIMPLE_MODEL_MODEL REPORTS ####################################
REP	DRT	
	TYPE	VOLTAGE CELL 1
	CELLS	SIMPLE MODEL COLUMN layer SIMPLE MODEL SIMPLE MODEL 1 somaE
	PROB	1
	REPORT ON	VOLTAGE
	FILENAME	SIMPLE_MODEL_1_VOLTAGE_E.txt
	ASCII	
	FREQUENCY	1
	TIME_START	Θ
	TIME_END	100
END_	REPORT	
REP)RT	
	ТҮРЕ	VOLTAGE CELL 2
	CELLS	SIMPLE MODEL COLUMN laver SIMPLE MODEL SIMPLE MODEL 2 somaE
	PROB	1
	REPORT ON	VOLTAGE
	FILENAME	SIMPLE MODEL 2 VOLTAGE E.txt
	ASCII	
	FREQUENCY	1
	TIME_START	Θ
	TIME_END	100
END_	REPORT	

####	+######################################	SIMPLE_MODEL_MODEL REPORTS ####################################
REPO)RT	
	ТҮРЕ	VOLTAGE CELL 1
	CELLS	SIMPLE MODEL COLUMN laver SIMPLE MODEL SIMPLE MODEL 1 somaE
	PROB	1
	REPORT ON	VOLTAGE
	FILENAME	SIMPLE MODEL 1 VOLTAGE E.txt
	ASCII	
	FREQUENCY	1
	TIME_START	Θ
	TIME_END	100
END_	REPORT	
REPO)RT	
	ТҮРЕ	VOLTAGE CELL 2
	CELLS	SIMPLE MODEL COLUMN laver SIMPLE MODEL SIMPLE MODEL 2 somaE
	PROB	1
	REPORT ON	VOLTAGE
	FILENAME	SIMPLE MODEL 2 VOLTAGE E.txt
	ASCII	
	FREQUENCY	1
	TIME START	Θ
	TIME_END	100
END_	REPORT	

#######################################	SIMPLE_MODEL_MODEL REPORTS ####################################
REPORT TYPE CELLS PROB REPORT_ON FILENAME ASCII FREQUENCY TIME_START	<pre>VOLTAGE_CELL_1 SIMPLE_MODEL_COLUMN layer_SIMPLE_MODEL SIMPLE_MODEL_1 somaE 1 VOLTAGE SIMPLE_MODEL_1_VOLTAGE_E.txt 1 0</pre>
TIME_END END_REPORT	100
REPORT TYPE CELLS PROB REPORT_ON FILENAME ASCII FREQUENCY TIME_START TIME_END END_REPORT	VOLTAGE_CELL_2 SIMPLE_MODEL_COLUMN layer_SIMPLE_MODEL SIMPLE_MODEL_2 somaE 1 VOLTAGE SIMPLE_MODEL_2_VOLTAGE_E.txt 1 0 100

##############	##### SIMPLE_MODEL_MODEL REPORTS ####################################
REPORT	
TYPE	VOLTAGE CELL 1
CELLS	SIMPLE MODEL COLUMN layer SIMPLE MODEL SIMPLE MODEL 1 somaE
PROB	1
REPORT ON	VOLTAGE
FILENAME	SIMPLE MODEL 1 VOLTAGE E.txt
ASCII	
FREQUENCY	1
TIME_STAR	ΓΘ
TIME_END	100
END_REPORT	
DEDODT	
	STMPLE MODEL COLUMN lavor STMPLE MODEL STMPLE MODEL 2 comaE
PROB	1
REPORT ON	VOL TAGE
	SIMPLE MODEL 2 VOLTAGE E. txt
ASCII	
FREQUENCY	1
TIME STAR	ΓΟ
TIME_END	100
END_REPORT	

#######################################	SIMPLE_MODEL_MODEL REPORTS ####################################
REPORT	
TYPE	VOLTAGE_CELL_1
CELLS	SIMPLE_MODEL_COLUMN layer_SIMPLE_MODEL SIMPLE_MODEL_1 somaE
PROB	1
REPORT_ON	VOLTAGE
FILENAME	SIMPLE_MODEL_1_VOLTAGE_E.txt
ASCII	
FREQUENCY	1
TIME START	Θ
TIME_END	100
END_REPORT	
DEDODT	
TTPE	VULIAGE_CELL_Z
CELLS	SIMPLE_MODEL_COLOMN layer_SIMPLE_MODEL SIMPLE_MODEL_2 somae
PROB	
REPORT_ON	
FILENAME	SIMPLE_MODEL_2_VOLIAGE_E.TXT
ASCII	
FREQUENCY	1
TIME_START	0
TIME_END	100
END_REPORT	

#######################################	SIMPLE_MODEL_MODEL REPORTS ####################################
REPORT	
TYPE	VOLTAGE CELL 1
CELLS	SIMPLE MODEL COLUMN laver SIMPLE MODEL SIMPLE MODEL 1 somaE
PROB	1
REPORT ON	VOLTAGE
FILENAME	SIMPLE MODEL 1 VOLTAGE E.txt
ASCII	
FREQUENCY	1
TIME START	Θ
TIME END	100
END_REPORT	
REPORT	
TYPE	VOLTAGE_CELL_2
CELLS	SIMPLE_MODEL_COLUMN layer_SIMPLE_MODEL SIMPLE_MODEL_2 somaE
PROB	1
REPORT_ON	VOLTAGE
FILENAME	SIMPLE_MODEL_2_VOLTAGE_E.txt
ASCII	
FREQUENCY	1
TIME_START	Θ
TIME_END	100
END_REPORT	

#######################################	SIMPLE_MODEL_MODEL REPORTS ####################################
REPORT	
TYPE	VOLTAGE CELL 1
CELLS	SIMPLE MODEL COLUMN layer SIMPLE MODEL SIMPLE MODEL 1 somaE
PROB	1
REPORT ON	VOLTAGE
FILENAME	SIMPLE MODEL 1 VOLTAGE E.txt
ASCII	
FREQUENCY	1
TIME_START	Θ
TIME_END	100
END_REPORT	
DEDODT	
	VOLTAGE CELL 2
CELLS	SIMPLE MODEL COLUMN laver SIMPLE MODEL SIMPLE MODEL 2 somaE
PROB	
REPORT ON	VOLTAGE
FILENAME	SIMPLE MODEL 2 VOLTAGE E.txt
ASCII	
FREQUENCY	1
TIME START	Θ
TIME_END	100
END_REPORT	

Output Analysis

Graphing

- Tools:
 Matlab
 GNUplot
- Types of plots

 Dot and Line graphs
 Raster plots
 Spectrogram

Types of Plots


Types of Plots



Types of Plots







Today's Outline

• First Hour

- o Introduction
- Equations and Implementation
- o Requirements and Simulation on a Single Machine
- o Input Language

Second Hour

- o Simple Model
- Parameters Presentation and Testing
- o Output Analysis

Third Hour

- o Simulation on Multiple Machines
- o Software Tools
- o Robotic System Configuration
- o Larger Networks and Complete Loop Execution
- o Future Directions and Summary

CPU

- A single unit of execution (Core)
 - o Often times sold with multiple cores
- A single instruction executed once per cycle per core
 - o i.e. add X and Y
- Most of the silicon in the chip devoted to:
 - o Branch Handling
 - o Cache and Memory controllers
 - o Out of order execution
 - o etc.
- Design optimized for general preformance

CPU Layout

3rd Generation Intel[®] Core[™] Processor: 22nm Process



New architecture with shared cache delivering more performance and energy efficiency

Quad Core die with Intel[®] HD Graphics 4000 shown above Transistor count: 1.4Billion Die size: 160mm² "Cache is shared across all 4 cores and processor graphics

GPU

- Groupings of 32 simple cores
- Single instruction executed 32 times per cycle
 i.e. add X_i Y_i
- Most of the silicon is devoted to ALUs(Arithmetic Logic Units)
- Design optimized for parallelism and floating point math performance

GPU Layout



CPU vs GPU

- Access to Memory
 - o CPU much closer to RAM and other memory
 - o GPU has onboard memory, but
- Programming model
 - o GPU much more suited to data parallel problems
 - typically image processing, graphics, matrix multiplication
 - Very array centric
 - Avoids pointer manipulation and branching
 - o CPU much more suited to general computing problems
- Raw floating point performance
 - o CPU 100 GFLOPS (i7 980 XE)
 - o GPU 1300 GFLOPS (GTX 480)
 - o Both from 2010

MPI

- Message Passing Interface
- Handles the dirty details of networking
 - o Endianness
 - o Managing sockets
 - o Grouping Nodes
- Provides many methods for sending data out
 - o Single Node to Single Node (Send Receive)
 - o Single Node to Many Nodes (Scatter)
 - o Many Nodes to Single Node (Gather)
- Designed for use in high performance networks

Simulation on multiple machines

One-Time Step

- SSH keys allow password free access to all computers.
 - o ssh <computerName>
 - ssh-keygen -t rsa accept default options
 - o cd ~/.ssh
 - o cp id_rsa.pub authorized_keys

One-Time Step

- cd /home/userName/NCS6/NCS6/build
- Create a file with .mpi extension. This file specifies the number of devices available on each computer in the cluster. For example, we have marbles.mpi file that contains the following information:

Brain1 slots=2

Brain2 slots=2

- After creating a file with .mpi extension, run these commands:
 - mpirun --hostfile marbles.mpi <space> applications/clusterSpecifier/clusterSpecifier <space> marbles.cluster
 - o applications/clusterInfo/clusterInfo marbles.cluster

Steps

• To compile code:

applications/ncsDistributor/ncsDistributor <space> ../files/NCS6/ marbles.cluster ncsout

• To run code:

mpirun -np numberOfDevices -hostfile
<space> marbles.mpi <space>
applications/simulator/simulator ncsout/



Software Tools

NeuroTranslate

NeuroTranslate

 Software tool that translates input files between NCS and NeuroML

😣 🖨 💷 Neuro Translate			
File Help			
Brain Columns Layers	Cells Connections Synapses Stimuli Report	:s	
		A	
Туре	brain_model_input		
Job	brain_model_output		
FSV	1000 Hz	-	
Duration	10 s	-	
Seed	-20 -int		
Column Types	k ⁱ Random number generator. visual_cortex hypothalamus parietal_cortex		
	Add/Remove		
Stimulus Injects	visual proprioceptive		
No file imported	NCS	NeuroML	

N. Jordan, K. Perry, N. Narala, L. C. Jayet Bray, and F. C. Harris, Jr. Design and implementation of an NCS-NeuroML translator. In Proceedings of the International Conference on Software Engineering and Data Engineering (SEDE). Los Angeles, CA, June 2012.

🛇 🗇 🐵 gridTest.xml (~/workspace/NeuroTranslate/samples) - gedit	
File Edit View Search Tools Documents Help	
🔂 🛅 Open 👻 🏧 Save 🔛 🦣 Undo 🧀 💥 👘 🎁 🔍 😪	
gridTest.xml 🗱	
xml version="1.0" encoding="UTF-8"?	6
<br This example shows a Level 3 compliant file, containing a number of connected cell models >	(0)
<pre><neuroml length_units="micrometer" level3="" neuroml_level3_v1.8.1.xsd"="" v1.8.1="" xmlns="http://morphml.org/neuroml/schema" xmlns:bio="http://morphml.org/biophysics/schema" xmlns:cml="http://morphml.org/channelml/schema" xmlns:meta="http://morphml.org/metadata/schema" xmlns:mml="http://morphml.org/morphml/schema" xmlns:net="http://morphml.org/networkml/schema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemalocation="http://morphml.org/neuroml/schema" xsi:schemata=""> </neuroml></pre>	
<pre><meta:notes>A completely specified network in NeuroML Level 3. While this is useful for exporting/importing/saving from an application, better practice might be to have the cell definitions, the channel mechanisms, and network specification in separate files.</meta:notes></pre>	
The cell types present in the network	
<cells> <cell name="CellA"> <meta:notes>Test cell for showing how channels can be placed on a cell</meta:notes></cell></cells>	
<pre><segments xmlns="http://morphml.org/morphml/schema"></segments></pre>	•
XML 👻 Tab Width: 8 👻 Ln 14, Col 155 INS	



Robotic System Configuration

Virtual NeuroRobotic (VNR)



- 1. The robot is sufficiently embodied for the human to tentatively accept the robot as a social, emotional partner
- 2. The human-robot interaction loop operates in real time, with no pre-specified parcellation into receptive and responsive time windows
- 3. The cognitive control is a neuromorphic brain emulation incorporating realistic neuronal dynamics with time constants that reflect synaptic activation and learning, established membrane and circuitry properties
- 4. The neuromorphic architecture can potentially provide circuitry underlying intrinsic motivation and intentionality, using "emotional" rather than rulebased learning & reinforcement
- 5. The neuromorphic architecture is expandable to progressively larger scale and complexity to support brain model development and validation



Robot

Filter

NCS

- Models integrate-and-fire neurons with conductance-based synapses
- First simulator to support real-time neurorobotics applications
- Experiments demonstrate biologically realistic behavior in real time

Server

- Brain Communication Server (BCS)
- Monitors the robotic avatar and creates the appropriate stimulus for proprioceptive feedback and premotor movement to replicate the role of a biological brainstem

NCSTools

NCSTools

 Software package that simplifies interaction and communication between NCS and remote agents



C. M. Thibeault, J. Hegie, L. Jayet Bray, and F. C. Harris, Jr. Simplifying neurorobotic development with nestools. In Proceedings of the 2012 Conference on Computers and Their Applications. Las Vegas, NV, March 2012.

Visual / Audio

- Computer vision / audio
- Machine vision / audio
- Image / sound processing
- Filtering mechanisms (e.g. Gabor)
- Applications:
 - o external input
 - o reward-based learning

Robotic Interface



- Constructed using Webots 5
- Motions were programmed in C++ using the provided interfaces and the communication was accomplished using the NCSTools C++ client

Large Networks

Technical Approach



Brain Model



Trust

- Behavior between a humanoid neurorobot and human actor
 - o Oxytocin release
 - Social reinforcement
 - Reduction of inhibition
- Experiment has two conceptual phases:
 - o Learning
 - Neurorobot initiates a sequence of motions
 - Human performs concordant or discordant action
 - Neurorobot learns to trust the human
 - o Challenge
 - Human reaches for another object
 - Depending on whether or not the neurorobot trusts the human the robot will hand over the object or retract the object

Trust and Affiliation

Capuchin Monkeys Display Affiliation Toward Humans Who Imitate Them

Annika Paukner,¹* Stephen J. Suomi,¹ Elisabetta Visalberghi,² Pier F. Ferrari^{1,3} SCIENCE VOL 325 14 AUGUST 2009

A B C



Willingness to exchange token for food

Paradigm

LEARNING

Robot Initiates Action

1. Robot brain initiates arbitrary sequence of motions







("match"), or different ("mismatch") pattern

Human Responds

Match: robot Mismatch: learns to trust don't trust



CHALLENGE (at any time)

Human Acts

3. Human slowly reaches for an object on the table

Robot Reacts

4. Robot either "trusts", (assists/offers the object), or "distrusts", (retract the object).





distrusted







 \rightarrow





Microcircuitry



Video Input – Gabor Filtering

- Images are processed and values are sent to the simulated visual pathways (V1, V2 and V4)
- Input closely resembles how visual information is processed in a biologically realistic brain




Trust the Intent Recognition Discordant Motions



Trust the Intent Recognition Concordant Motions





Discordant > Distrust





Concordant > Trust





Navigation

Navigate to familiar location

- o Prefrontal Cortex
- o Hippocampus
- o Subiculum
- Computional system representing a navigating rodent
- Reward at the end of a sequence of turns
- Showed learning performance without biased decisions
- Short-term memory

Paradigm



Microcircuitry







Emotional Speech

- Allows for more natural interaction between humans and robots
 - o Determine the ideal behavior from a simple reward feedback
- Emotional Speech processor
 - o Successfully distinguished "sad" and "happy" utterances
- Integrated into neurorobotic scenario

 The robot received a spoken reward if the correct decision was made
- Neurorobot successfully and consistently learned
 the exerceise
- Step toward the combination of human emotions and virtual neurorobotics

REWARD-BASED LEARNING THROUGH ESP



L. C. Jayet Bray, G. Ferheyhough, E. Barker, C. M. Thibeault, P. H. Goodman, and F. C. Harris, Jr.. Emotional speech processing in neurorobotics. In revision, 2012.

ESP CLASSIFICATION PERFROMANCE

TABLE IHUMAN CLASSIFICATION CONFUSION MATRIX

Category	Anger	Fear	Нарру	Sad	Error
Anger	62	3	5	0	11.4%
Fear	5	62	1	2	11.4%
Нарру	5	8	56	1	20.0%
Sad	0	1	1	68	2.9%
Average Error					11.4%

ESP RECOGNITION PERFROMANCE

TABLE II OFFLINE MODE RECOGNITION CONFUSION MATRIX

Category	Happy-M	Sad-M	Happy-F	Sad-F	Error
Нарру-М	16	0	0	0	0.0%
Sad-M	2	13	0	0	13.3%
Happy-F	0	0	17	1	5.6%
Sad-F	0	0	0	12	0.0%
Average Error					4.7%

ESP RECOGNITION PERFROMANCE

TABLE III LIVE MODE RECOGNITION CONFUSION MATRIX

Category	Happy-M	Sad-M	Happy-F	Sad-F	Error
Нарру-М	22	0	0	0	0.0%
Sad-M	0	16	0	0	0.0%
Happy-F	0	0	19	1	5.0%
Sad-F	0	0	0	19	0.0%
Average Error					1.3%







Complete Loop Execution

Requirements

Save

o NCS files (NCS_core)

 Configuration files (Reward_Based_Learning)

Robotic files (Webots_Neighborhood)

Folders in home directory

Steps

Start voServer on port 20003

- cd /home/username/NCS_core/voServer
- o ./server <space> -p <space> 20003

Open volnterpreter

- o cd /home/username/NCS_core/ncstools/bin
- o ./volnterpreter <space>
 /home/username/Reward_Based_Learning/input/navigation.cfg

Start NCS

- o cd /home/username/Reward_Based_Learning
- o ./ncs5e 1 ./input/navigation.in

Steps

Video Capture

o cd

/home/username/Reward_Based_Learning/card_color_detection

o ./recognize_card

Start webots and load world

o webots



Future Directions

Future Directions

- Multi-Scale/Mixed Models:
 - o Izhikevich and NCS and ... all in the same model
- Published Interface for New Neuron/Synapse Models
 - Allow your own coding of neurons and synapses and use our parallel code.
- Speed....
 - o Always here ☺
- More Parameters on NCS Neurons/Synapses
- Visualization: 2D and 3D

Future Directions

- Research into Memory:
- Tools:
 - o GUI Brain Builder,
 - o Output Analysis
- ModelDB
- Input language options
 o PyNN (like)



First Hour

- Introduction
- NCS history and development
- Current enhancements
- Equations and Implementation
- Software and hardware requirements
- How to run a small model on a single machine
- Overview of the input language

Second Hour

- Detailed description of available parameters
- Demos
- Output analysis

Third Hour

- CPU, GPU, and MPI
- How to run on multiple machines
- Software tools
- Robotic system configuration
- Large scale models
- Complete loop execution
- Future directions

Acknowledgments

Office of Naval Research



DARPA Synapse project and HRL





Brain Computation Laboratory

University of Nevada, Reno







Laurence Jayet Bray Fred Harris, Jr Sergiu Dascalu Director

in funded collaboration with U de Cergy-Pontoise and CNRS, Paris, France University of Bonn, Germany Brain Mind Institute (Blue Brain Project), EPFL, Lausanne, Switzerland



Mathias Quoy René Doursat Florian Morman Henry Markram Jim King



Oral session IV: Navigation

- Monday July 23
- 10:40 11:00
- Talk: 012
- Goal-Related Navigation of a Neuromorphic Virtual Robot
 - Laurence Jayet Bray, Emily Barker, Gareth Ferneyhough, Roger Hoang, Bobby Bryant, Sergiu Dascalu, and Frederick C Harris

Workshop 4

- Multi-Scale Modeling in Computational Neuroscience II: Challenges and Opportunities
 - o Wed: 9-6



Brain Computation Lab

http://www.cse.unr.edu/brain/



Brain Computation Lab

Navigation

- Research Projects
- People
- Publications
- Sponsors
- Conferences
- Opportunities
- University of Nevada, Reno
- Department of Computer
 Science and Engineering
- School of Medicine
- Biomedical Engineering Program



Welcome to the Brain Laboratory!

Good Afternoon!

Founded in 2001, the brain lab is a joint research center between the departments of Computer Science & Engineering, Medicine, Physiology & Cell Biology, and the program of Biomedical Engineering. It also has neurobiological collaborations with the Brain Mind Institute at the EPFL (Switzerland), the University of Cergy Pontoise (France), and the University of Bonn (Germany).

Our researchers consists primarily of undergraduate/graduate students and alumni of the University of Nevada, Reno. They are actively developing computational innovations to understand the physiological processes that give rise to neocortical memory, learning, and cognition. Our models and experiments help understand brain pathophysiology and create brain-like artificial intelligence and neural prosthetic devices.

New Publications

- Design and Implementation of an NCS-NeuroML Translator
- Real-Time Human-Robot Interaction Underlying Neurorobotic Trust and Intent Recognition
- Correlation Maps Allow Neuronal Electrical Properties to be Predicted from Singlecell Gene Expression Profiles in Rat Neocortex
- Heterogeneity in the Pyramidal Network of the Medial Prefrontal Cortex