

FRANKLIN & MARSHALL

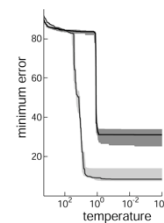
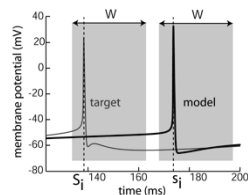


## Parameter Optimization of Neuron Models: Lessons from the Field

**Christina M. Weaver**

Department of Mathematics & Computer Science  
Franklin & Marshall College  
Lancaster, PA

NEURON Users Group Meeting, Tucson, AZ  
March 23, 2010



## Acknowledgments

- Experimental collaborators: Bob Baker (NYU Med School), Jennifer Luebke (Boston U. Sch. Med.)
- Andrew Davison (CNRS)

## Susan L. Wearne (1955-2009)



## A common starting point

Develop a model to reproduce electrophysiologic data.

$$\text{Neuron} + \int \frac{1}{C_m} \sum I_{ion} dt = \text{Action Potential Graph}$$

To do this, we must find appropriate parameter values. But how?

- Hand tuning takes time.
- The model has many interacting parameters.
- Are the resulting parameters best, in some quantitative way?

**Automated parameter search is a useful tool...**

**...but automated search must be used carefully.**

- Need a fitness (“objective”) function to represent essential differences between model and experimental (“target”) data.
- Many different search methods can be used to optimize the fitness function.
- Which fitness function, and which search method, is most appropriate depends on the data you want to reproduce.

**This workshop will provide an introduction to parameter optimization, and demonstrate optimization tools available in NEURON.**

## Outline of the Workshop



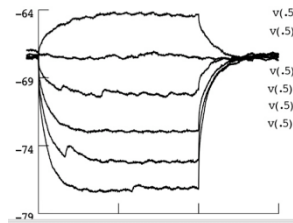
- Optimization Overview:
  - Typical Data & Model
  - Fitness function examples
  - Common search methods
  - Limitations of parameter fitting
- Optimization with NEURON
  - Intro to the Multiple Run Fitter
  - Fitting subthreshold data
  - Fitting suprathreshold data
  - Interacting with MATLAB
- General discussion

## Typical Experimental Data

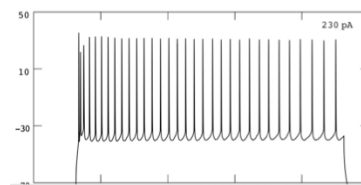
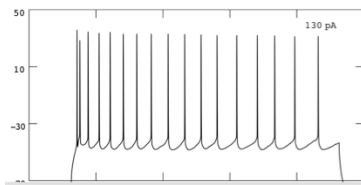
Morphology



Subthreshold Physiology



Suprathreshold Physiology (APs)



These are our target data.

## A mathematical model based on Hodgkin & Huxley

$$C_m \frac{dV}{dt} = \bar{g}_{Na} m^3 h (V - V_{Na}) + \bar{g}_K n^4 (V - V_K) + g_L (V - V_L)$$

$$\frac{dm}{dt} = \alpha_m(V)(1-m) + \beta_m(V)m \quad \frac{dh}{dt} = \alpha_h(V)(1-h) + \beta_h(V)h$$

$$\frac{dn}{dt} = \alpha_n(V)(1-n) + \beta_n(V)n \quad \frac{dC}{dt} = K_p \cdot (-I_{Ca}) - R \cdot C$$

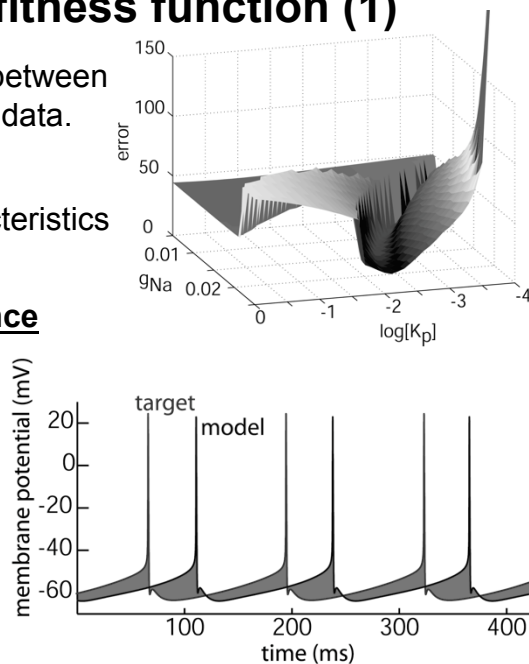
- Active ion channels, plus leak
- Equations for activation/inactivation variables (channel kinetics)
- Other ion channels ( $\text{Ca}^{2+}$ ,  $\text{Ca}^{2+}$ -dependent  $\text{K}^+$ ) and  $[\text{Ca}^{2+}]_i$  often included.

## Choosing a fitness function (1)

- To quantify differences between model output and target data.
- Often: want spike times, adaptation, spike characteristics

- Mean Squared Difference** between voltage traces:

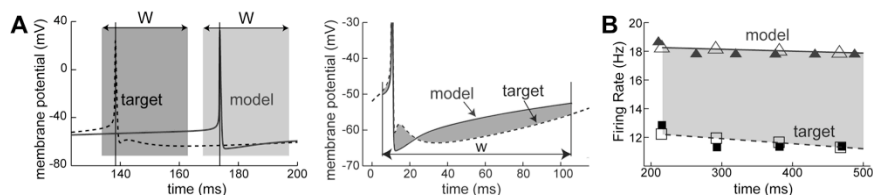
Good for subthreshold data, but not when there are APs.



## Choosing a fitness function (2)

- Action Potential Shape and Time-Varying Firing Rate:** (Weaver & Wearne '06; Weaver & Wearne '08)

$$E(M) = \alpha E_{AP} + \beta E_{FR\ slope} + \gamma E_{FR\ fit}$$

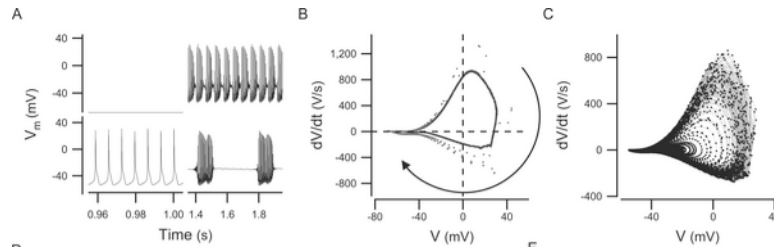


- “Peel off” model APs and align them with model APs; then compute shape error
- Compute target & model firing rates, fit them each to a line. *Intercept*: baseline FR; *Slope*: simple measure of adaptation.

## Choosing a fitness function (3)

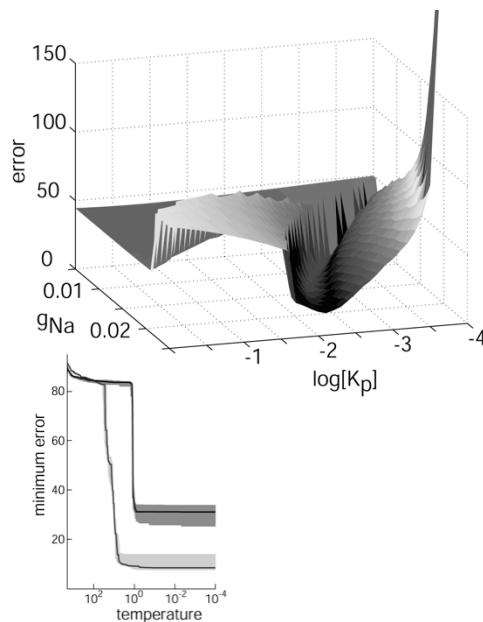
- **Phase Plane Trajectory Density**

(LeMasson & Maex '00; Achard & De Schutter '06;)



- Plot  $dV/dt$  vs.  $V$ . Break phase plane into grid; compute density of target points in each grid block, vs. model density.
- Removes time dependence, can give great results.
- In my models: didn't fit data as well as my APshp + FR function.

## Choosing a Search Method (1)



### Praxis (standard in NEURON)

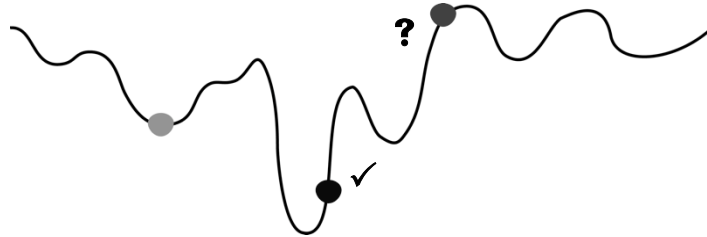
- Conjugate gradient method
- From an initial point, walk "downhill" to the local minimum.
- But **global** minimum?

## Choosing a Search Method (2)

### Simplex-Based Simulated Annealing

(Nelder & Mead '65; Kirkpatrick et al. '83; Press et al. '92):

- Simplex wanders around parameter space, seeking the global optimum.



- Downhill, always accept.
- Uphill, probability proportional to “temperature”.
- Temperature starts high, decreases slowly.
- Works for constrained problems (Cardoso et al. '96, Weaver & Wearne '06).



## Choosing a Search Method (3)

### Genetic Algorithms (Holland '75; Eiben & Smith '03)

- Search consists of a series of generations, each derived from parents of the previous generation.
- “Best” individuals of the current generation become parents of the next generation, via crossover.
- Mutation, random variation in parameters, can also occur.
- Has been very successful, in neuron models & elsewhere, but (I think) not publicly available in NEURON.

## Limitations of Parameter Optimization

- Fits are only as good as your data. (Poor data = poor model!)
- Lack of experimental constraints (e.g. dendritic voltage, kinetics)
- Choosing weights of different components
- Choosing 'meta-parameters' of the search method
- With real data, an "acceptable model" may not exist!

**Still, it's a powerful alternative/complement to hand tuning.**

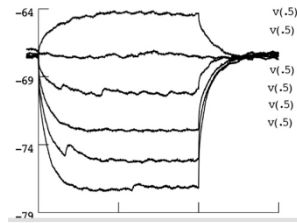
## Outline of the Workshop



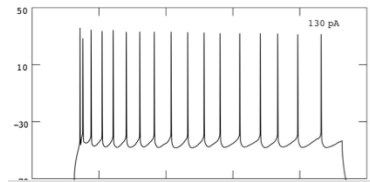
- Optimization Overview:
  - Typical Data & Model
  - Fitness function examples
  - Common search methods
  - Limitations of parameter fitting
- Optimization with NEURON
  - Intro to the Multiple Run Fitter
  - Fitting subthreshold data
  - Fitting suprathreshold data
  - Interacting with MATLAB
- General discussion

## In practice: what to use, when?

Subthreshold



Suprathreshold (APs)



### MSD with Praxis

- ~Linear → local min OK
- Including ion channels active subTH (e.g.  $g_H$ ,  $g_A$ ): SA/GA may be needed

### APshp+FR, PPTD, ... with

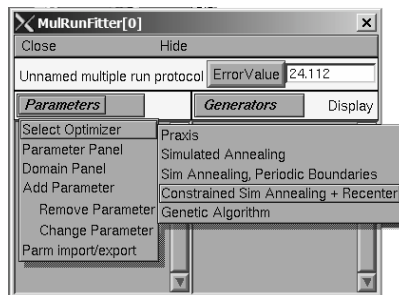
**SimAnn:** for single, global minimum

or

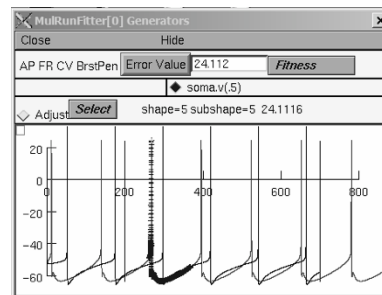
**GenAlg:** for several, near-global minima

## NEURON's Multiple Run Fitter

Set up parameters + data to fit



Specify Fitness Function



## NEURON's MRF tutorial:

<http://www.neuron.yale.edu/neuron/static/docs/optimiz/main.html>

- **Set up “Run Fitness” and specify data to fit**
  - Add protocol constant
  - Use Generator → ON
- **Choose parameters**
- **Choose fitness method**
  - Search results written to \*.tmp, \*.fit files
- **Sample: subthreshold fit of g\_pas and Cm.**

## For suprathreshold data: (Weaver & Wearne '06, '08, ModelDB)

<http://senselab.med.yale.edu/senselab/modeldb/ShowModel.asp?model=87473>

(An update reflecting today's presentation is at  
<http://www.neuron.yale.edu/ftp/hines/tucson>)

- unix/linux/mac: type the following at the command line:
  - > cd weaver\_NRN2010\_OptmzDemo
  - > nrnivmodl model optmz
  - > nrngui mosinit\_passive.hoc
- mswin: start nrngui
  - Use mknrndll in the optmz & model directories
  - File -> working dir to first the model
  - File -> working dir to the optmz directory.
  - File -> load hoc and select mosinit\_passive.hoc.
  - (Also: save .dll's in weaver\_NRN\* directory, changing names as needed)

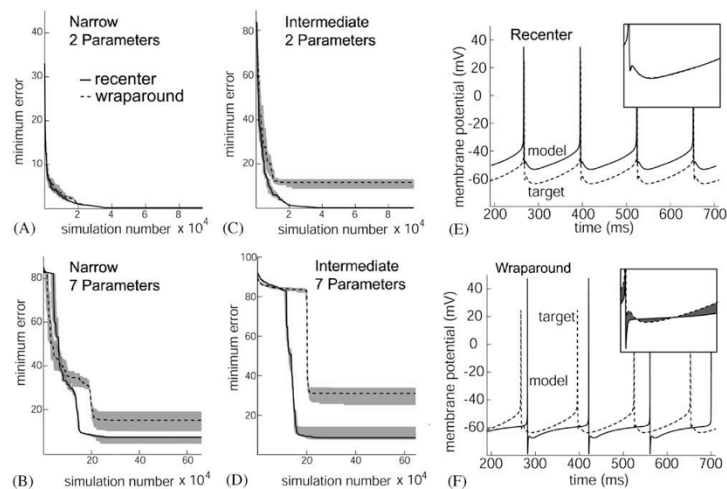
## For suprathreshold data: (Weaver & Wearne '06, '08, ModelDB)

- Several sets of parameters, comparing RegionFitness (MSD) and my APshp/FRtime function

	MSD	APshp/FRtime
Target	6.9e-8	.002
Slow FR	77.36	78.76
High FR	118.39	42.66
Shallow AP	92.93	67.22
Deep AP	66.54	42.32
Burst	161.03	646.56
No spikes	56.33	96.06

## For suprathreshold data: (Weaver & Wearne '06, ModelDB)

- Simulated Annealing: Still quite heuristic. It can be slow, but it can find good fits to data.

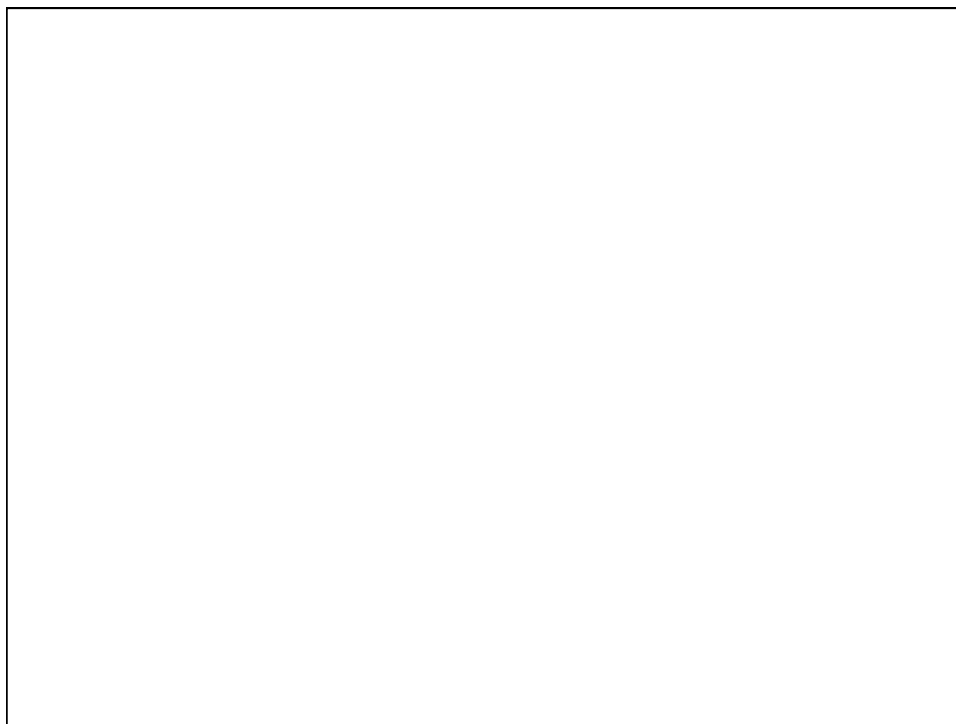


## Genetic Algorithms with MATLAB:

- You can call NEURON within MATLAB:  
`system('/Applications/NEURON-7.1/nrn/i686/bin/nrngui tmp.hoc')`
- Or on WINDOWS:  
`dos('c:\nrn61\bin\innriv.exe -nogui tmp.hoc')`
- Genetic Algorithm in MATLAB:  
`ga(@fitnessfcn,nvars,A,b,Aeq,beq,LB,UB)`
- **May be superior to Simulated Annealing, especially to find several “good” fits (even if not “best”).**

## References: Parameter Optimization for Neuronal Models

1. Achard, P. and E. De Schutter, Complex parameter landscape for a complex neuron model. *PLoS Comput. Biol.* **2**: e94, (2006).
2. Druckmann, S., T.K. Berger, S. Hill, F. Schurmann, H. Markram, and I. Segev, Evaluating automated parameter constraining procedures of neuron models by experimental and surrogate data. *Biol Cybern* **99**(4-5): 371-9, (2008).
3. Eiben, A.E. and J.E. Smith, *Introduction to Evolutionary Computing*. Berlin Heidelberg New York: Springer-Verlag. (2003).
4. Keren, N., N. Peled, and A. Korngreen, Constraining compartmental models using multiple voltage recordings and genetic algorithms. *J. Neurophysiol.* **94**(6): 3730-3742, (2005).
5. Keren, N., D. Bar-Yehuda, and A. Korngreen, Experimentally guided modelling of dendritic excitability in rat neocortical pyramidal neurones. *J Physiol* **587**(Pt 7): 1413-37, (2009).
6. LeMasson, G. and R. Maex, Introduction to equation solving and parameter fitting, in *Computational Neuroscience: Realistic Modeling for Experimentalists*, E.D. Schutter, Editor. CRC Press: Boca Raton, FL. (2000).
7. Major, G., Passive modeling: a practical introduction, in *Computational Neuroscience: Realistic Modeling for Experimentalists*, E.D. Schutter, Editor. CRC Press: Boca Raton, FL. (2000).
8. Surkis, A., C.S. Peskin, D. Tranchina, and C.S. Leonard, Recovery of cable properties through active and passive modeling of subthreshold membrane responses from laterodorsal tegmental neurons. *J. Neurophysiol.* **80**(5): 2593-2607, (1998).
9. Trevelyan, A.J. and J. Jack, Detailed passive cable models of layer 2/3 pyramidal cells in rat visual cortex at different temperatures. *J. Physiol.* **539**(Pt 2): 623-636, (2002).
10. Vanier, M.C. and J.M. Bower, A comparative survey of automated parameter-search methods for compartmental neural models. *J. Comput. Neurosci.* **7**(2): 149-171, (1999).
11. Van Geit, W., E. De Schutter, and P. Achard, Automated neuron model optimization techniques: a review. *Biol Cybern* **99**(4-5): 241-51, (2008).
12. Weaver, C.M. and S.L. Wearne, The role of action potential shape and parameter constraints in optimization of compartment models. *Neurocomputing* **69**: 1053-1057, (2006).
13. Weaver, C.M. *AP shape and parameter constraints in optimization of compartment models*. ModelDB 2007; Available from: <http://senselab.med.yale.edu/senselab/modeldb/ShowModel.asp?model=87473>.
14. Weaver, C.M. and S.L. Wearne, Neuronal firing sensitivity to morphologic and active membrane parameters. *PLoS Comput. Biol.* **4**(1): 130-150, (2008).



### **Questions for Discussion:**

- Your experience with parameter fitting / optimization?
- Network level, or subcellular, parameter fitting?
- Things that might be useful?

**christina.weaver@fandm.edu**