Network tuning by genetic algorithms

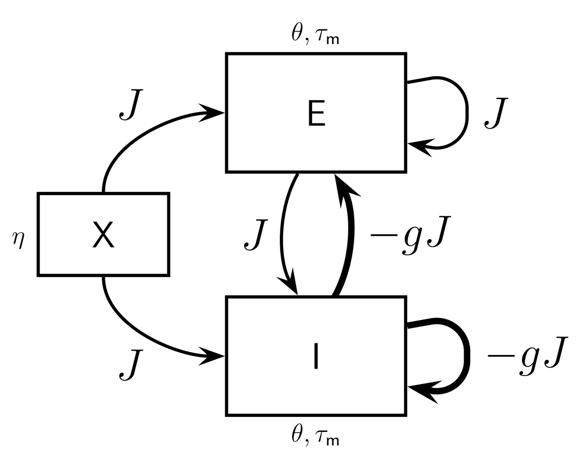
Håkon Enger, Tom Tetzlaff, and Gaute T. Einevoll

Dept. of Mathematical Sciences and Technology (IMT),
Norwegian University of Life Sciences (UMB), Ås, Norway

Introduction

- Neuronal network parameters:
  - network architecture
  - dynamics of synapses and single cells
- Biologically realistic systems:
  - high-dimensional parameter space
  - constrained only to some extent
- Genetic algorithms provide a potential solution to this problem
- Questions:
  - To what extent can network parameters be determined by fitting the population statistics of neural activity?
  - What's a good fit strategy?
  - What's the precision of the fit result?
  - How important are individual parameters for the functional performance of the model?

Balanced Random Network Model



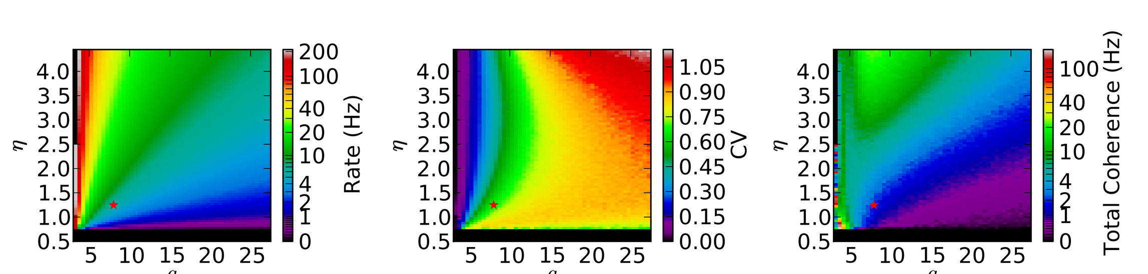
A	**Network model:** Multi-population random network with fixed in-degrees (Brunel, 2000)
Populations	E (excitatory): LIF neurons (see **B**), size N_E
	I (inhibitory): LIF neurons (see **B**), size N_I
	X (external): Poisson point processes with rate $\eta\nu_\theta$, size $K_E(N_E + N_I)$
Connectivity	EE, IE: Random convergent $K_E \to 1$, excitatory synapses
	EI, II: Random convergent $K_I \to 1$, inhibitory synapses (see **C**)
	EX, IX: Non-overlapping $K_E \to 1$, excitatory synapses (see **C**)
Parameters	Population sizes $N_{\{E,I\}}$, in-degrees $K_{\{E,I\}} = \epsilon N_{\{E,I\}}$, connectivity ϵ, **relative external drive** η

B	**Neuron model:** Leaky integrate-and-fire (LIF) neuron (Lapicque, 1907; Tuckwell, 1988)
Spike emission	Neuron $k \in [1, N_E + N_I]$ fires at all times $\{t_{j_k} \vert V_k(t_{j_k}) = \theta,\ j_k \in \mathbb{N}\}$
Subthreshold dynamics	$\tau_m \dot{V}_k = -V_k + RI_k(t)$ if $\forall j_k : t \notin (t_{j_k}, t_{j_k} + \tau_{ref}]$ Total synaptic input current $I_k(t) = \sum_l \sum_{j_l} i_{kl}(t - t_{j_l})$ (see **C**)
Reset + refractoriness	$V_k(t) = V_{reset}$ if $\forall j_k : t \in (t_{j_k}, t_{j_k} + \tau_{ref}]$
Parameters	Membrane time constant τ_m, membrane resistance R, **spike threshold** θ, reset potential V_{reset}, refractory period τ_{ref}

C	**Synapse model:** Static current synapse with α-function shaped PSC
PSC kernel	$i_{kl}(t+d) = \begin{cases} J_{kl} e \tau_s^{-1} t\, e^{-t/\tau_s} & t > 0 \\ 0 & \text{else} \end{cases}$
Synaptic weights	$J_{kl} = \begin{cases} J & \text{if synapse } kl \text{ exists and is excitatory} \\ -gJ & \text{if synapse } kl \text{ exists and is inhibitory} \\ 0 & \text{else} \end{cases}$
Parameters	**Excitatory synaptic weight** J, **relative strength** g **of inhibition**, **synaptic time constant** τ_s, synaptic delay d

D	**Spike-train analysis**
Spike trains	$s_k(t) = \sum_{j_k} \delta(t - t_{j_k})$
Population averaged firing rate	$r_0 = \langle s_k(t) \rangle_{k,t}$
Coefficient of variation of inter-spike interval	CV $= \left\langle \sqrt{\left\langle T_{j_k}^2 \right\rangle_{j_k} - \langle T_{j_k} \rangle_{j_k}^2} / \langle T_{j_k} \rangle_{j_k} \right\rangle_k$ with $T_{j_k} = t_{j_k+1} - t_{j_k}$
Population averaged spike-train coherence	$\kappa(\omega) = C(\omega)/P(\omega)$ with cross-spectrum $C(\omega) = \mathfrak{F}_\tau \left[\langle s_k(t) s_l(t+\tau) \rangle_{k,l\neq k,t} \right](\omega)$ and power-spectrum $P(\omega) = \mathfrak{F}_\tau \left[\langle s_k(t) s_k(t+\tau) \rangle_{k,t} \right](\omega)$
Total coherence	$\kappa_0 = \int_{-\omega_{max}}^{\omega_{max}} d\omega\ \kappa(\omega) \approx \Delta\omega \sum_{\omega=-\omega_{max}}^{\omega_{max}} \kappa(\omega)$

Description of the model and the spike-train analysis. Blue-marked parameters are varied during the optimisation.

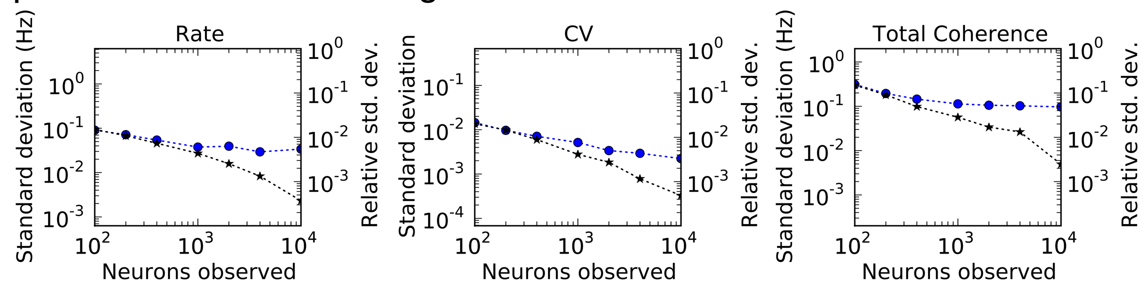


Average rate, CV and total coherence for the 2-dimensional (g, η) parameter space, other parameters kept constant. ⋆ marks reference point $g = 8$, $\eta = 1.25$, $J_{psp} = 0.1$mV, $\theta = 20$ mV, $\tau_s = 0.01$ms.

Variability of measures

- Sources of variability in measured states:
  - due to recording from only a selection of neurons in the network
  - due to statistical fluctuations in network structure

The statistical fluctuations may be studied by varying the seed of the pseudorandom number generator used to construct the network.



● Varying random seed ★ Varying observed neurons

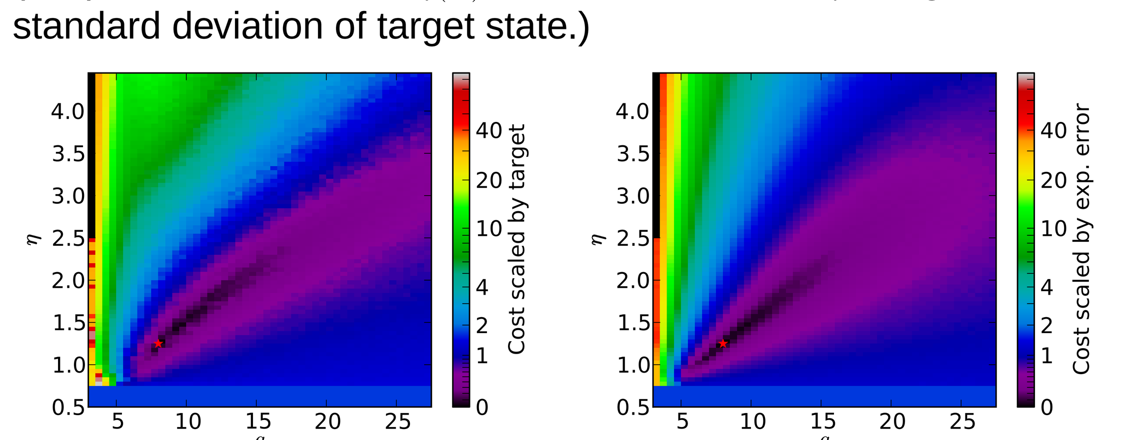
Alternative cost functions

The cost function used in the optimization algorithm should be dimensionless. There are two natural ways of achieving this:

– Scaling by target value: $f(\mathbf{p}) = \sum_i (s_i(\mathbf{p}) - t_i)^2 / t_i^2$

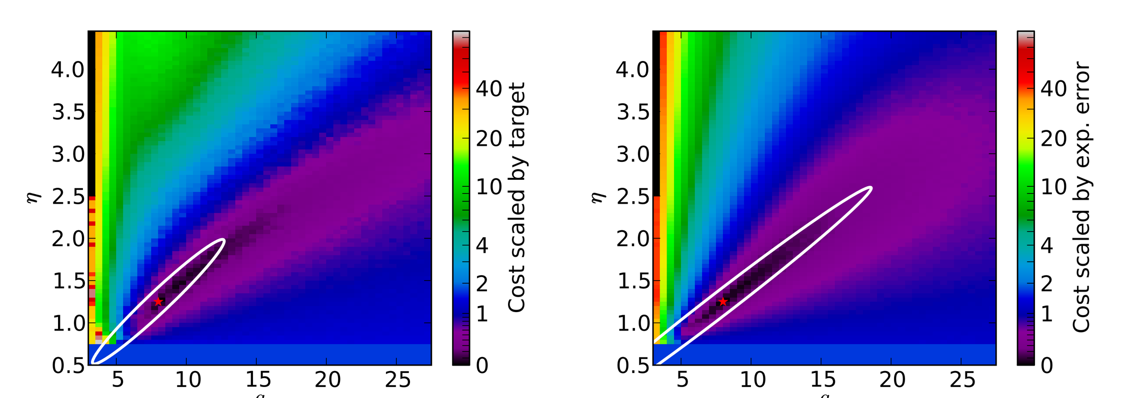
– Scaling by natural target variability: $f(\mathbf{p}) = \sum_i (s_i(\mathbf{p}) - t_i)^2 / \Delta t_i^2$

($\mathbf{p}$: parameter vector, $s_i(\mathbf{p})$: measured state, t_i: target state, Δt_i: standard deviation of target state.)



Local cost landscapes

- Observation: shallow valley near minimum.
  - Indicates insensitivity to certain parameter combinations. (Gutenkunst et al., 2007)
- Exploration of full cost landscape not practical for high-dimensional parameter spaces.
- Instead, find the local curvature of the cost landscape by studying the Hessian $H_{jk} = \partial^2 f / \partial p_j \partial p_k$ of the cost function.



Ellipses show isocontour lines of quadratic approximation of cost function found by analysis of the Hessian (Gutenkunst et al., 2007).

Because of the intrinsic statistical fluctuations of the measured quantities, the data used to calculate the Hessian is noisy. The noise is amplified when calculating derivatives, since this involves subtracting two data points of equal magnitude. In order to obtain the Hessian used for the above plot, we averaged over 200 different network realizations in each point.

Acknowledgements

We acknowledge support by the NOTUR and eScience programs of the Research Council of Norway. All network simulations were carried out with the neural simulation tool NEST (see http://www.nest-initiative.org).

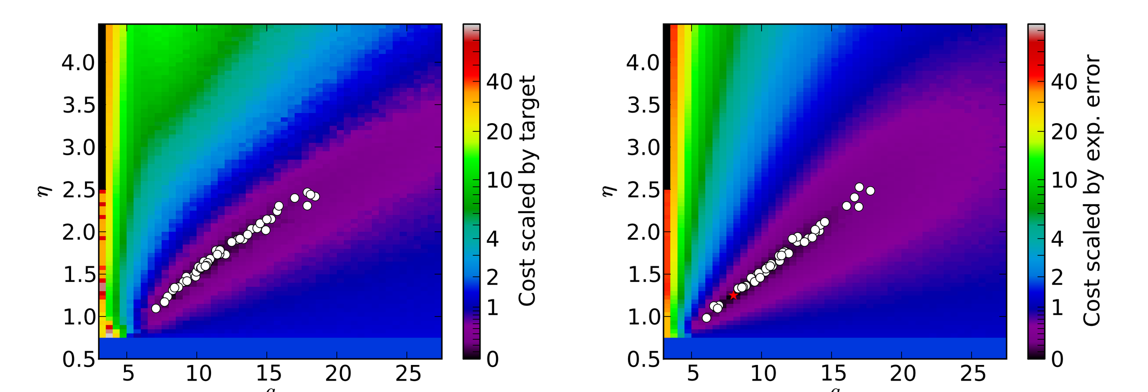
Genetic algorithm

- Evaluating the cost function is very time expensive.
- Must use a minimal number of individuals per generation and a strategy which converges quickly to the minimum.

We use a genetic algorithm with 20 individuals in each generation, and generational replacement except for an elitist rule where the best individual from the previous generation is kept. We employ roulette wheel selection with linear ranking and crossover mating. The mutation probability is 0.01 per bit and the selective pressure is 2.0.

- Results from repeated searches using different initial generation:

2-dimensional parameter space

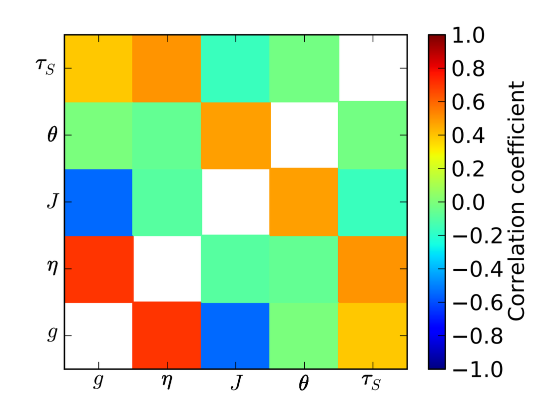


Figures show the best result from 50 optimizations using a genetic algorithm.

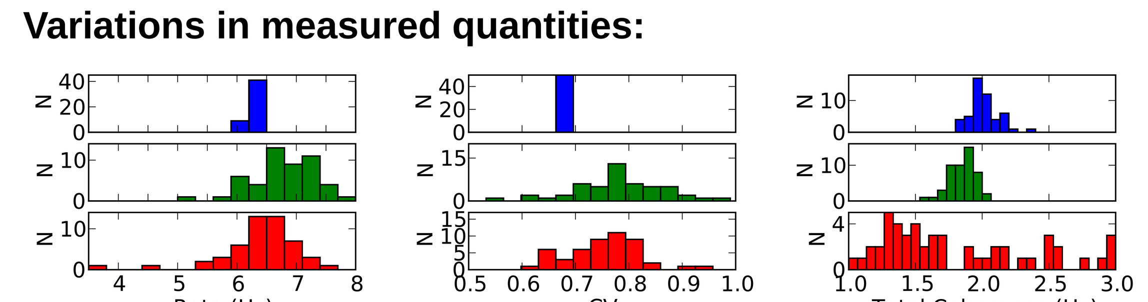
5-dimensional parameter space

Results and correlations:

Parameter	Result
g	14 ± 5
η	2.3 ± 0.6
J_{psp}	0.2 ± 0.1 mV
θ	29 ± 8 mV
τ_s	0.5 ± 0.2 ms

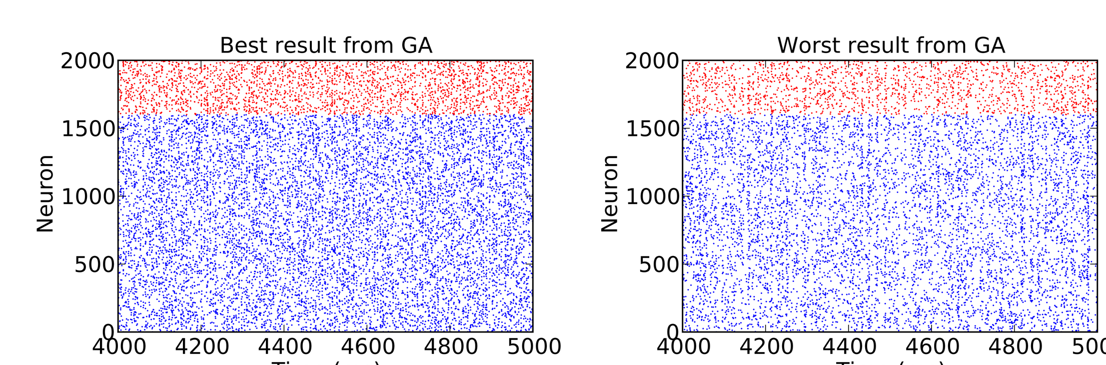


Variations in measured quantities:



Blue: Target variability; **Green:** Best results from genetic algorithm using cost function scaled by target value; **Red:** Best results using cost function scaled by experimental error.

- Comparing network activity from "best" and "worst" result of optimization:



Conclusion

- Intrinsic variability in network state
  - ⇒ Inevitable variability in fitted parameters
- Different observables have different precision
- Cost landscape in vicinity of minima often shallow along certain directions, i.e. similar performance for different parameter combinations
- Choice of cost function affects the accuracy of the genetic algorithm

References

Brunel N (2000). Dynamics of sparsely connected networks of excitatory and inhibitory spiking neurons. *J Comput Neurosci* 8(3): 183–208

Gutenkunst RN, et al. (2007). Universally Sloppy Parameter Sensitivities in Systems Biology Models. *PLoS Comput Biol* 3(10): e189

Lapicque L (1907). Recherches quantitatives sur l'excitation électrique des nerfs traitée comme une polarisation. *J Physiol Pathol Gen* 9: 620–635

Tuckwell HC (1988). *Introduction to Theoretical Neurobiology*, vol. 1 (Cambridge University Press)

NORWEGIAN UNIVERSITY OF LIFE SCIENCES MDCCCLIX