Managing heterogeneity in the study of neural oscillator dynamics

Carlo R Laing*[1] , Yu Zou[2] , Ben Smith[1,3] and Ioannis G Kevrekidis[2]

[1]Institute of Information and Mathematical Sciences, Massey University, Private Bag 102-904 North Shore Mail Centre, Auckland 0632, New Zealand
[2]Department of Chemical and Biological Engineering and Program in Applied and Computational Mathematics, Princeton University, Princeton, NJ, 08544, USA
[3]Research Centre for Cognitive Neuroscience,Department of Psychology, University of Auckland, Private Bag 92019, Auckland 1142, New Zealand

*Corresponding author: c.r.laing@massey.ac.nz

Email: CRL: c.r.laing@massey.ac.nz
YZ: yzou@princeton.edu
BS: benjsmith@gmail.com
IGK: yannis@princeton.edu
;

Corresponding author

Abstract

We consider a coupled, heterogeneous population of relaxation oscillators used to model rhythmic oscillations in the pre-Bötzinger complex. By choosing specific values of the parameter used to describe the heterogeneity, sampled from the probability distribution of the values of that parameter, we show how the effects of heterogeneity can be studied in a computationally efficient manner. When more than one parameter is heterogeneous, full or

ideal, and thus, it is more realistic to consider **heterogeneous** networks. Also, there is evidence in a number of contexts that heterogeneity within a population of neurons can be beneficial. Examples include calcium wave propagation [18], the synchronisation of coupled excitable units to an external drive [19, 20], and the example we study here: respiratory rhythm generation [13, 21].

One simple way to incorporate heterogeneity in a network of coupled oscillators is to select one parameter which affects the individual dynamics of each oscillator and assign a different value to this parameter for each oscillator [3, 15, 22, 23]. Doing this raises natural questions such as from which distribution should these parameter values be chosen, and what effect does this heterogeneity have on the dynamics of the network?

Furthermore, if we want to answer these questions in the most computationally efficient way, we need a procedure for selecting a (somehow) optimal representative set of parameter values from this distribution. In this paper, we will address some of these issues.

In particular, we will show how - given the distribution(s) of the parameter(s) describing the heterogeneity - the representative set of parameter values can be chosen so as to accurately incorporate the effects of the heterogeneity without having to fully simulate the entire large network of oscillators.

We investigate one particular network of coupled relaxation oscillators, derived from a model of the pre-Bötzinger complex [13, 14, 24], and show how the heterogeneity in one parameter affects its dynamics. We also show how heterogeneity in more than one parameter can be incorporated using either full or sparse tensor product grids in parameter space.

$P_N(1) = 1$, and the weights

$$w_i = \quad 1$$

Hopf bifurcation results in a canard periodic solution [32] which very rapidly increases in amplitude as $\mathbf{I_m}$ is increased. This is related to the separation of time scales between the $\mathbf{V}$ dynamics (fast) and the $\mathbf{h}$ dynamics (slow). In the left panel of Figure 6, we see that some of the neurons in the network whose

function with a bounded second derivative, but the inverse CDF of a normal distribution (i.e. $Q^{-1}(z)$) does not have a bounded second derivative, and an error analysis of Equation 22 (not shown) predicts a scaling of M^{-1}, as observed.

6 Sparse grids

The process described in the previous section can obviously be generalised to more than two randomly, but independently, distributed parameters. The distribution of each parameter determines the type of quadrature which should be used in that direction, and the parameter values and weights are formed from tensor products of the underlying one-dimensional rules. However, the curse of dimensionality will restrict how many random parameters can be accurately sampled. If we use N points in each of D random dimensions, the number of neurons we need to simulate is N^D.

One way around this problem is to use sparse grids [33, 34], as introduced by Smolyak [35]. The basic idea is to use sparse tensor products, chosen in such a way as to have similar accuracy to the corresponding full tensor product, but with fewer grid points, and thus (in our case) fewer neurons to simulate. A general theory exists [33, 34], but to illustrate the idea, suppose we have two uncorrelated random parameters, each is distributed uniformly between -1 and 1. A full tensor product for the Gauss-Legendre quadrature using 11 points in each direction is shown in Figure 13.

To form a two-dimensional sparse grid using the Gauss-Legendre quadrature, we first write the one-dimensional integration rule for integrating a function f as

$$\int_{-1}^{1} f(x)\,dx \approx U^{i}(f) \equiv \sum_{j=1}^{N_i} w_j f(x_j), \tag{27}$$

where $i \in \mathbb{N}$; w_j are the weights, and x_j are the nodes. We form a nested family of such rules with index i where the correspondence between i and N_i is given in the following:

i.e. $N_i = 2^{i+1} - 1$. Then, the level L rule in two spatial dimensions is

$$A(L, 2) = \sum_{|i|=L} \left(U^{i_1} \otimes U^{i_2}\right) - \sum_{|i|=L-1} \left(U^{i_1} \otimes U^{i_2}\right), \tag{28}$$

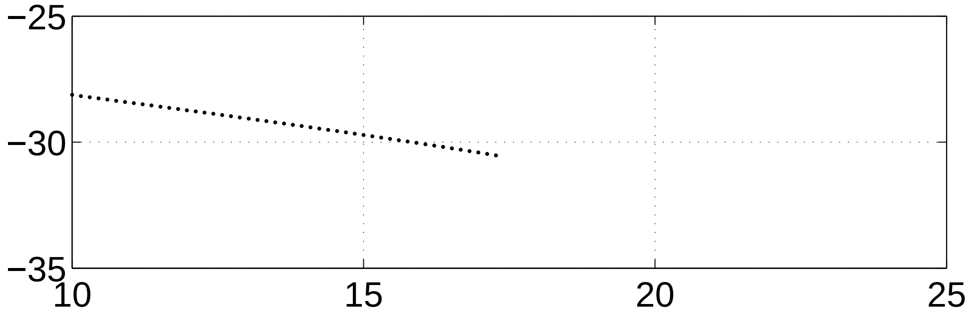
where $i \in \mathbb{N}^2$ and $|i| = i_1 + i_2$. The approximation of the integral of f over the domain $[-1, 1]^2$ is $A(L, 2)(f)$. So for example, the level 2 rule (in 2 spatial dimensions and using Gauss-Legendre quadrature) is

$$\begin{aligned} A(2, 2) \quad = \quad & U^0 \otimes U^2 + U^1 \otimes U^1 + U^2 \otimes U^0 \\ & - \left(U^0 \otimes U^1 + U^1 \otimes U^0\right) \end{aligned} \tag{29}$$

The grid for this rule is shown in Figure 14 (top), along with grids corresponding to several of its components.[a] Figure 14 (bottom) shows the grid for rule $A(3, 2)$.

Rules such as these can be constructed for an arbitrary number of spatial dimensions, using a variety of quadrature rules (and possibly different rules in different dimensions). Their advantage becomes apparent as the dimension of the space to be integrated over (or in our case, the number of heterogeneous parameters) is increased. To illustrate this, we consider as an example the model given by Equations 1 and 2 with I_{app} uniformly spread between 17.5 and 32.5, the g_{Na} uniformly spread between 2.55 and 3.05, V_{syn1}

proach [40] in which the equations satisfied by the polynomial chaos coe cients are never actually derived. They also chose the heterogeneous parameter values randomly from a prescribed distribution and



−25
−30
−35
10
15
20
25

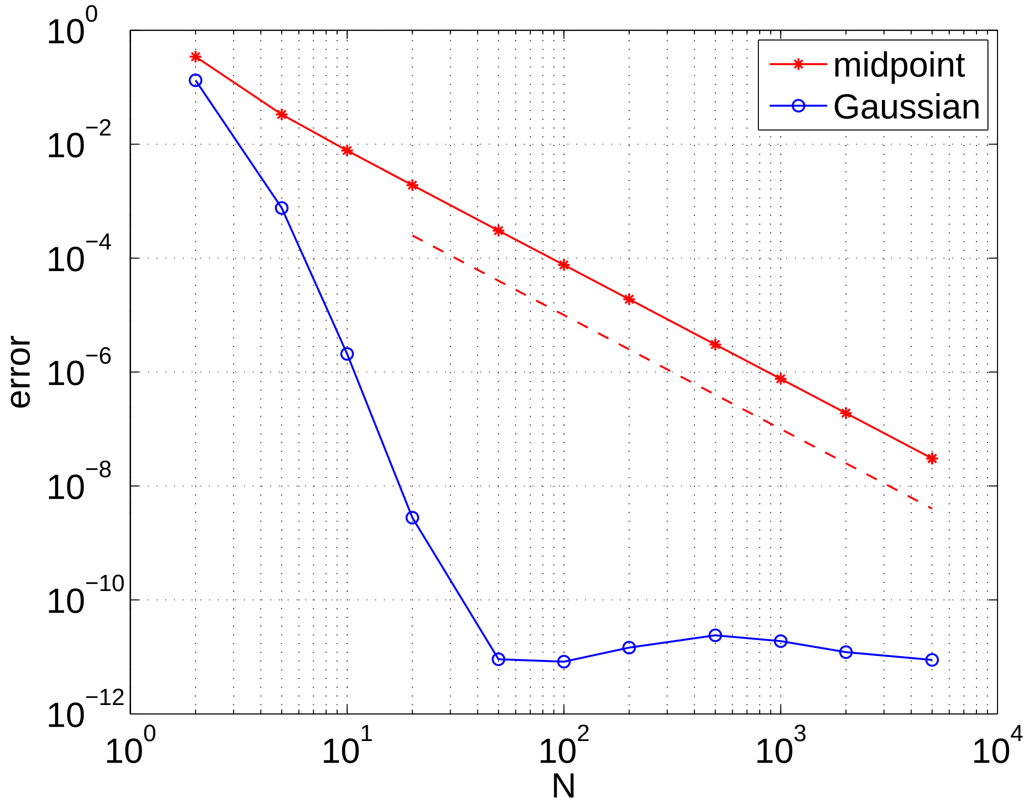
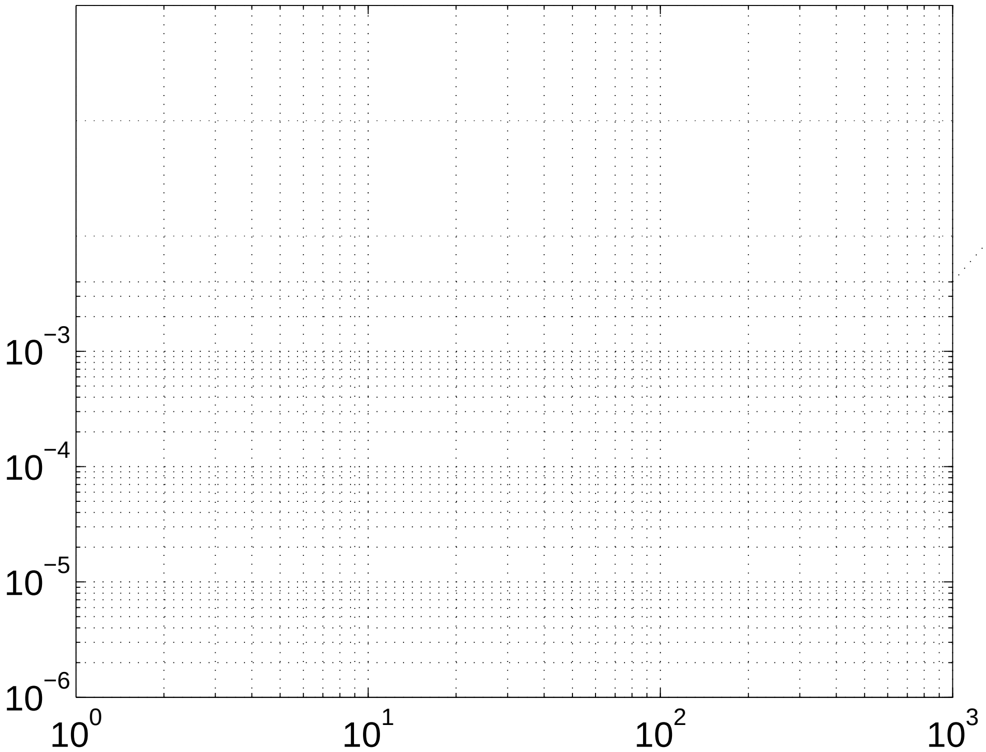
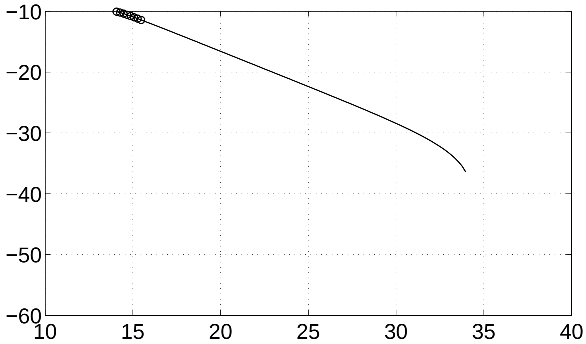
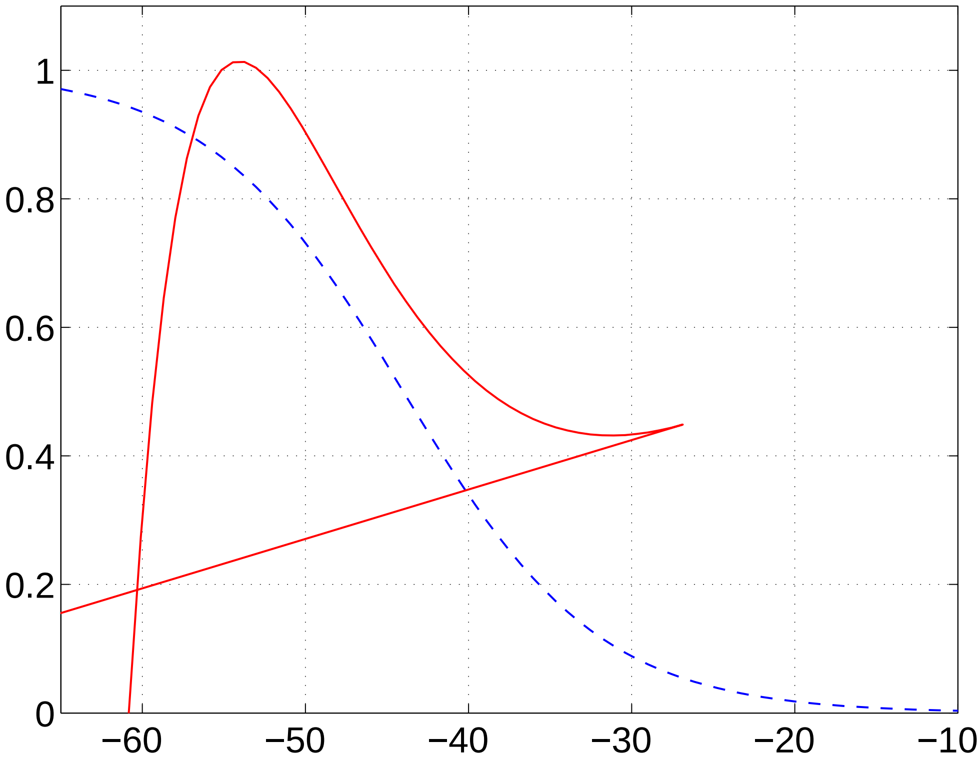


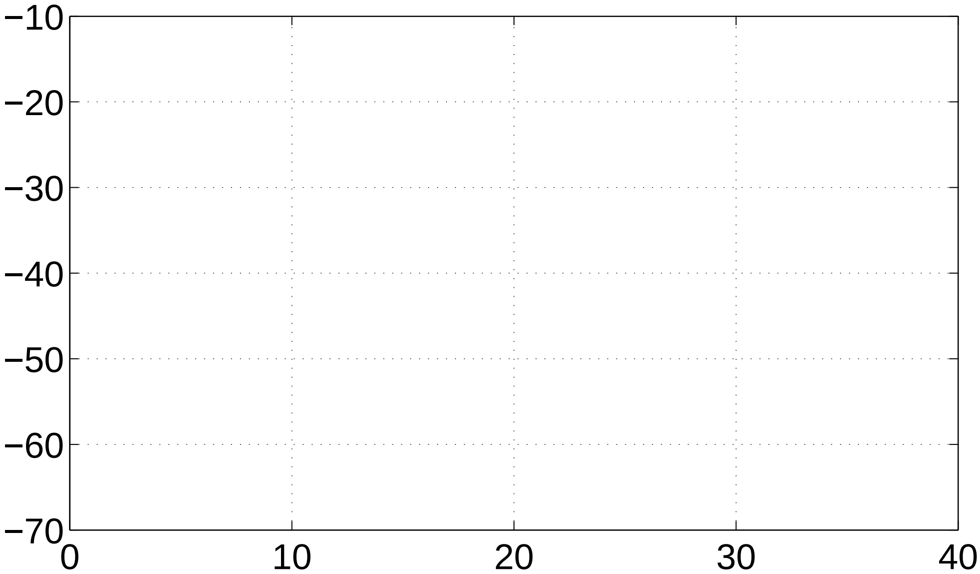
Figure 3: **Error in the calculated period of the synchronised oscillators.** Error in the calculated period of the synchronised oscillators as a function of the number of neurons simulated ($\mathbf{N}$) for the midpoint rule (red stars) and Gaussian quadrature (blue circles). Also shown (dashed) is a line corresponding to error scaling as $\mathbf{N}^{-2}$, to guide the eye.

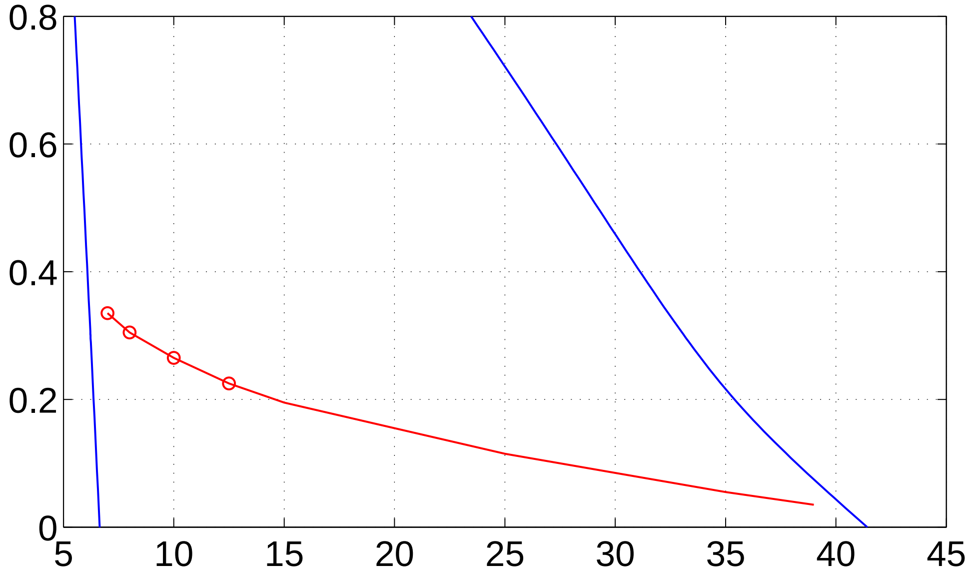

10^{-3}
10^{-4}
10^{-5}
10^{-6}
10^{0}
10^{1}
10^{2}
10^{3}

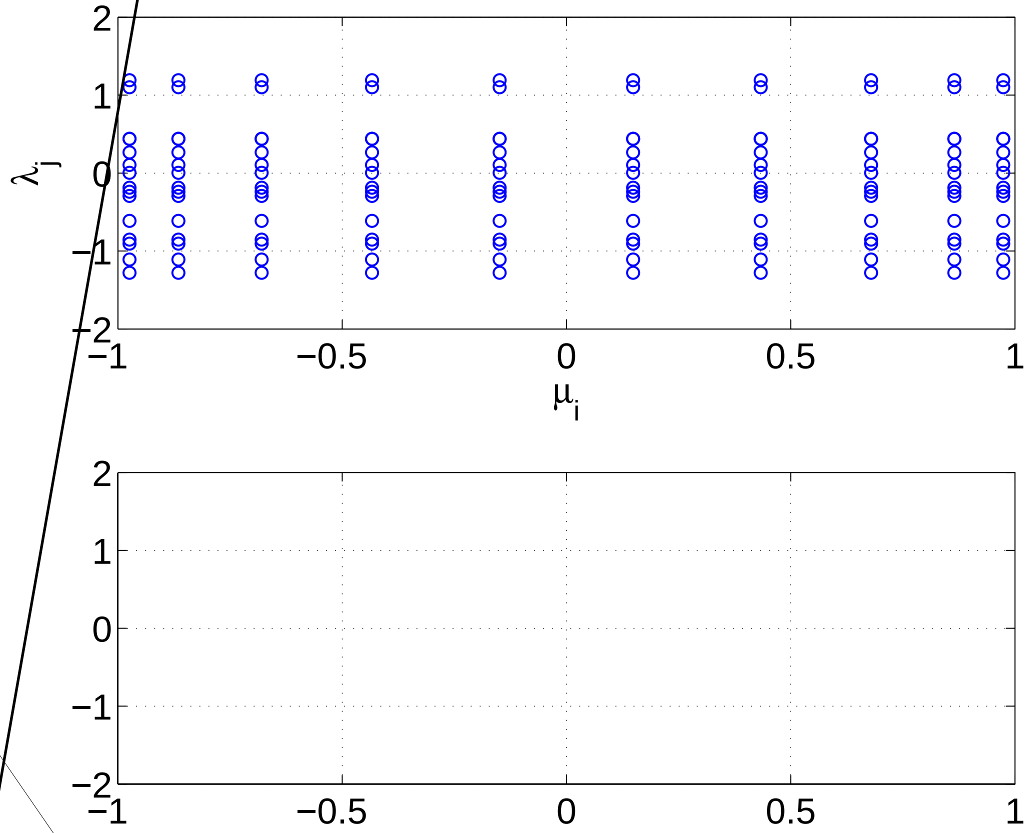

−10
−20
−30
−40
−50
−60
10
15
20
25
30
35
40



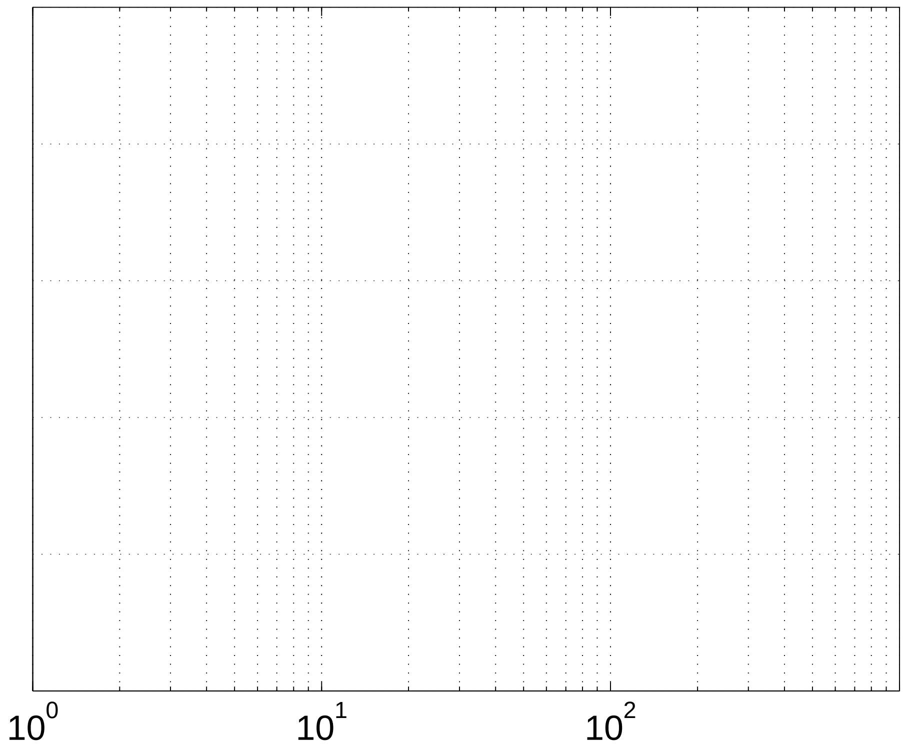
1
0.8
0.6
0.4
0.2
0
−60
−50
−40
−30
−20
−10


−10
−20
−30
−40
−50
−60
−70
0
10
20
30
40

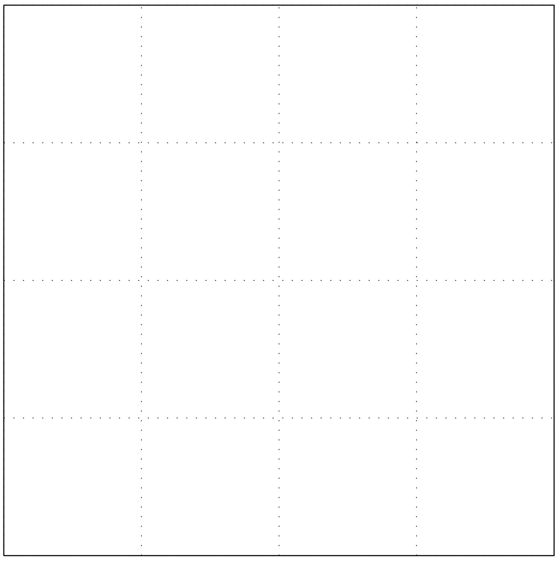

0.8
0.6
0.4
0.2
0
5
10
15
20
25
30
35
40
45



2
1
0
−1
−2
−1
−0.5
0
0.5
1
λj
μi
2
1
0
−1
−2
−1
−0.5
0
0.5
1



10^0
10^1
10^2



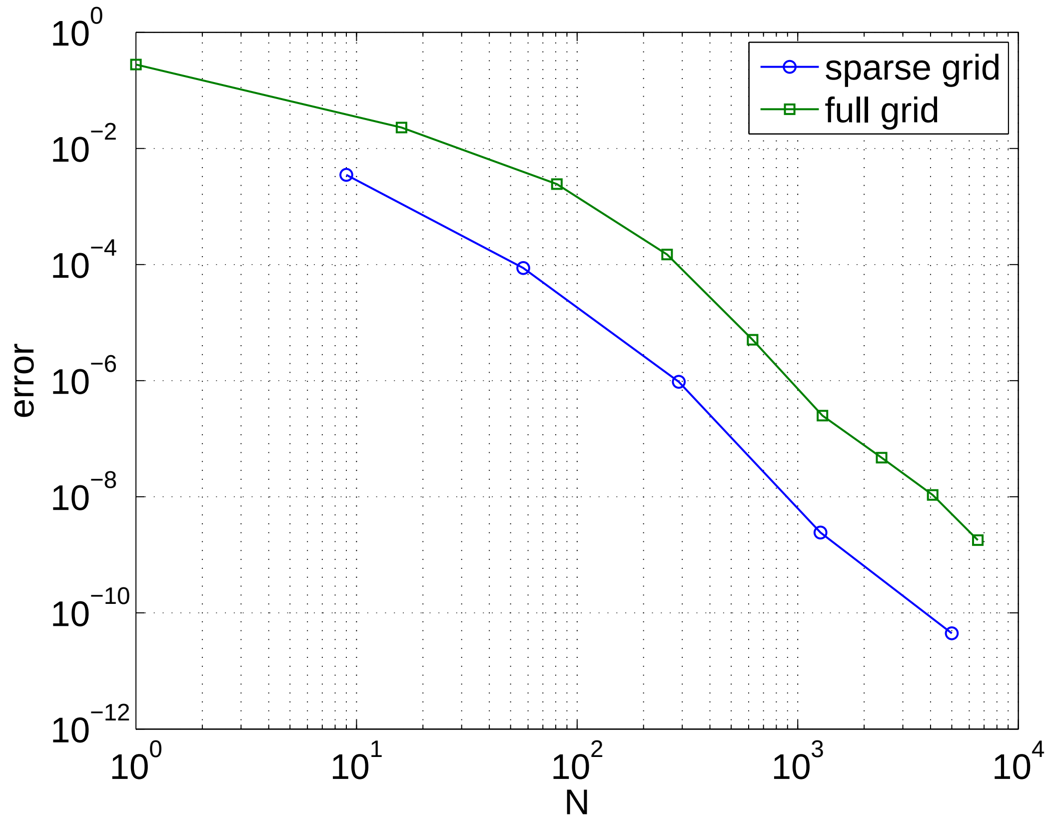


Figure 15: **Error in calculation of period.** Four distinct parameters are simultaneously heterogeneous (independently of one another) and we consider both full and sparse grids. See text for details. **N** is the number of neurons simulated.