

# IDENTIFICATION OF NEURONS PARTICIPATING IN CELL ASSEMBLIES

*Sonja Grün<sup>1,2</sup>, Denise Berger<sup>2,3</sup>, and Christian Borgelt<sup>4</sup>*

<sup>1</sup>Theoretical Neuroscience Group, RIKEN Brain Science Institute, Wako-shi, Japan

<sup>2</sup>Bernstein Center for Computational Neuroscience, Berlin, Germany

<sup>3</sup>Neuroinformatics, Inst. Biology, Freie Universität, Berlin, Germany

<sup>4</sup>European Center for Soft Computing, Mieres (Asturias), Spain

## ABSTRACT

Chances to detect assembly activity are expected to increase if the spiking activities of large numbers of neurons are recorded simultaneously. Although such massively parallel recordings are now becoming available, methods able to analyze such data for spike correlation are still rare, because it is often infeasible to extend methods developed for smaller data sets due to a combinatorial explosion. By evaluating pattern complexity distributions the existence of correlated groups can be detected, but their member neurons cannot be identified. In this contribution, we present approaches to actually identify the individual neurons involved in assemblies. Our results may complement other methods and also provide the opportunity for a reduction of data sets to the “relevant” neurons, thus allowing us to carry out a refined analysis of the detailed correlation structure due to reduced computation time.

**Index Terms**— massively parallel spike trains; spike synchrony; higher-order correlation; data mining

## 1. INTRODUCTION

Synchronized presynaptic spiking activity is known to have a higher efficacy in generating output spikes than non-coordinated spike timing [1]. Therefore temporal coordination of spike timing is a commonly accepted signature of neuronal assembly activity [3, 2, 4, 5]. Consequently, approaches to detect assembly activity have focused on the detection of correlated spiking activity on a millisecond time resolution.

With massively parallel recordings becoming available at an accelerating rate [6], chances to observe the signature of assembly activity are improving. However, currently we still lack the corresponding analysis tools [7]. Most of the existing methods are based on pairwise analysis, e.g. [8, 9, 10]. Approaches to analyze correlations between more than two neurons exist, but typically work only for a small number of neurons [11, 18, 12, 13] or consider only pair correlations when analyzing the ensemble [14, 15, 16, 17] (a set of neurons is seen as an assembly if most of them are pairwise correlated).

It is usually infeasible to simply extend existing methods that identify individual spike patterns to massively parallel

data due to a combinatorial explosion. Therefore we tried new approaches that evaluate the complexity distribution [19, 20] or the intersection matrix [21], which can handle massively parallel data and analyze it for higher-order spike patterns. These methods are able to detect the presence of higher-order correlation, but do not identify neurons that participate in the correlation. The goal of the present study is to resolve this issue: we want to directly identify neurons that take part in an assembly as expressed by coincident firing. Our aim is not, however, to determine the order of the correlation in which they are involved, but to provide an efficient tool to reduce the data set to the relevant neurons, which will then be examined in detail in further analysis. We present three different methods, all of which rely on the idea to detect whether an individual neuron is involved more often in any kind of coincident event than can be expected by chance.

## 2. GENERATION OF TEST DATA

In order to demonstrate our approaches we make use of four data sets of  $N = 100$  parallel spike trains, which contain different types of correlations or independent processes. The data sets are generated by a stochastic model, which has its origins in [22, 23, 20]. The basic assumption is that the activation of an assembly is expressed by synchronous spiking of its member neurons. Due to the typically blind sampling from the cortical tissue, the chances to observe a number of neurons from one assembly are rather small. This enters our modeling by assuming that only a small percentage of neurons are correlated—the rest fires independently.

### 2.1. Stochastic Model

Simultaneous spike trains are modeled as parallel, binary processes realized as stationary Bernoulli processes. The simplest form realizes fully independent processes with predefined firing rates  $\lambda_i$  per neuron  $i$ , thus defining the occupation probability  $p_i = \lambda_i \cdot h$  per time bin of length  $h$  for each process. Such realizations model the basic activity of the  $N$  neurons. Without further insertion of correlated spiking they serve as control data sets (“rate model”, Set1 in Fig. 1)

Assembly activity is modeled by coincident spiking activity in a subset of  $m$  out of the  $N$  neurons: a hidden “mother” process of rate  $\alpha$  is realized, from which spikes are copied into  $m$  selected child processes with probability  $\epsilon$ .  $\epsilon$  may be 1, so that all  $m$  processes receive a copy of a spike of the hidden process. In this case all  $m$  neurons exhibit coincidences of order  $m$  (Set2 in Fig. 1). Alternatively, and presumably more realistic for experimental data,  $\epsilon$  can be chosen  $< 1$ . In this case the resulting coincidences within the  $m$  neurons are on average composed of  $\epsilon \cdot m < m$  spikes with a random composition of spiking neurons per event (Set3 in Fig. 1).

Finally, the correlated and uncorrelated spike trains are merged. The spike train of a child process is then composed of “background” firing and of spikes involved in coincidences. The total firing rate is  $\lambda_i = \lambda_{i,b} + \alpha \cdot \epsilon$ , where  $\lambda_{i,b}$  is the background rate and  $\lambda_{i,c} = \alpha \cdot \epsilon$  the coincidence rate. Trivially, the firing rates of each process can be predefined, and the background firing rate can be adjusted accordingly.

Multiple assemblies can be generated analogously by using one hidden process per assembly. The sets of neurons, to which the spikes are copied from each of these processes, may overlap or not. The total rate of neurons taking part in more than one assembly is composed of the sum of the assembly coincidence rates and the background rate:  $\lambda_i = \lambda_{i,b} + \sum_j \lambda_{i,c_j}$  with assembly index  $j$  (Set4 in Fig. 1).

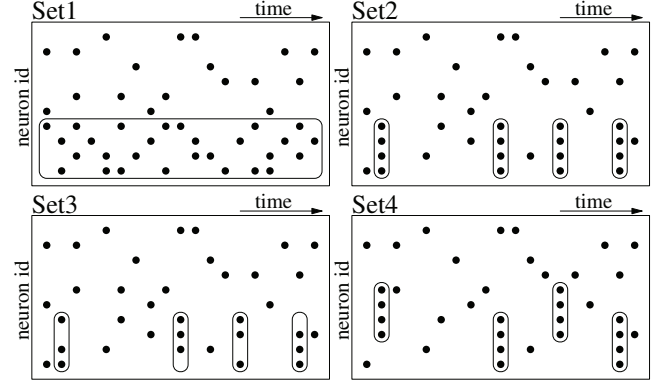
### 3. ANALYSIS OF ASSEMBLY MEMBERSHIP

We explored the performance of three approaches to identify whether a neuron is part of a correlated group of neurons.

#### 3.1. Spike Shuffling

In all three approaches explored here, we compute a different test statistic from the data, each of which is based on a different basic idea. Since there are certain obstacles to finding the distributions of these test statistics under the null hypothesis that the considered neuron is not part of an assembly, we rely on a spike shuffling procedure instead. First we compute, for a neuron  $i$  under consideration, the test statistic from the data. Then we shuffle the spike times of neuron  $i$  in order to make it independent of all other (observed) neurons, recompute the test statistic, and compare the results. Spike shuffling and recomputing the test statistic is repeated sufficiently often. Therefore counting the number of times the result of a shuffling trial meets or exceeds the result obtained on the actual data and dividing this number by the total number of trials yields a  $p$ -value. This  $p$ -value measures the probability that a result as obtained on the actual data would be obtained by chance if neuron  $i$  was independent of all other neurons.

Note that we shuffle *only* the spikes of neuron  $i$ ; all other spikes are left unchanged. As a consequence, the correlation structure among all other neurons is preserved—only correlations of neuron  $i$  with other neurons are destroyed.



**Fig. 1.** Sketches of the data sets used to test our methods. In the realizations, all of the  $N = 100$  neurons in Set2, Set3, and Set4 have a total firing rate of  $\lambda_i = 20\text{Hz}$ ,  $i \in [1 \dots N]$ ; coincidence firing rates are  $\lambda_c = 5\text{Hz}$ . Set1: rate model of independent firing, composed of two groups of neurons of different firing rates ( $\lambda_i = 50\text{Hz}$ ,  $i \in [1 \dots m]$ ,  $m = 10$  (box), and  $\lambda_i = 20\text{Hz}$ ,  $i \in [m + 1 \dots N]$ ). Set2 and Set3 contain one assembly of  $m = 10$  neurons generated with parameters  $[\epsilon = 1, \alpha = 5\text{Hz}]$  and  $[\epsilon = 0.4, \alpha = 12.5\text{Hz}]$ , respectively. Set4 contains two assemblies ( $\epsilon = 1$ ) with  $m_1 = 7$  and  $m_2 = 8$  neurons within  $m = 10$  neurons, 5 participate in both assemblies. Background firing rates have been chosen in such a way that the total rate of  $\lambda_i = 20\text{Hz}$  is preserved.

#### 3.2. Background Rate Estimation (BRE)

With the model outlined above, the probability that a neuron  $i$  fires in a given time bin may be decomposed into two constituents: the background firing probability, denoted by  $\theta_i = \lambda_{i,b} \cdot h$  (where  $h$  is the length of a time bin), and the coincidence probability, denoted by  $\xi_i = \lambda_{i,c} \cdot h$ , which captures the joint influence of *all* assemblies neuron  $i$  participates in. Since we assume that the hidden processes that generate the background and the coincident spikes are independent, the effective firing probability is  $\eta_i = \theta_i + \xi_i - \theta_i \xi_i$ .

We want to determine whether neuron  $i$  participates in an assembly or not, that is, whether  $\xi_i > 0$  (alternative hypothesis) or  $\xi_i = 0$  (null hypothesis). Due to the above equation we have  $\xi_i = \frac{\eta_i - \theta_i}{1 - \theta_i}$ . Note that  $\eta_i$  can easily be estimated from the data, namely as  $\hat{\eta}_i = k_i/k$ , where  $k_i$  is the number of time bins in which neuron  $i$  fires and  $k$  is the total number of time bins. As a consequence we can easily derive a statistical test if we can estimate the background firing probability  $\theta_i$ .

To derive the estimator for  $\theta_i$ , we consider a set of  $N$  independent neurons without assemblies (i.e.  $\forall j; 1 \leq j \leq N : \xi_j = 0$ ). The probability  $p_0$  that *no* neuron fires in a time bin is  $p_0 = \prod_{j=1}^N (1 - \theta_j)$  and the probability  $p_{i0}$  that *only* neuron  $i$  fires is  $p_{i0} = \theta_i \prod_{j=1, j \neq i}^N (1 - \theta_j)$ . It follows  $\frac{p_{i0}}{p_0} = \frac{\theta_i}{1 - \theta_i}$  and therefore  $\theta_i = \frac{p_{i0}}{p_{i0} + p_0}$ . Obviously,  $p_0$  and  $p_{i0}$  can easily be estimated, namely as  $\hat{p}_0 = k_0/k$  and  $\hat{p}_i = k_{i0}/k$ , respec-

tively, where  $k_0$  is the number of time bins in which *no* neuron fires and  $k_{i0}$  the number of bins in which *only* neuron  $i$  fires.

The crucial insight is now that the probabilities  $p_0$  and  $p_{i0}$  remain unaffected if neurons participate in assemblies, because both refer to time bins with *at most one* spike. However, coincident firing, by its very definition, means that *more than one* neuron fires in the same time bin. As a consequence,  $\theta_i$  can be estimated, even in the presence of coincident firing, as  $\hat{\theta}_i = k_{i0}/(k_{i0} + k_0)$ . The actual test whether neuron  $i$  participates in an assembly checks whether  $\hat{\xi}_i$  is sufficiently large so that the null hypothesis  $\xi_i = 0$  can be rejected. A natural test statistic, estimating the fraction of coincidence spikes, is

$$t_i^{\text{BRE}} = \frac{\hat{\xi}_i}{\hat{\eta}_i} = \frac{\hat{\eta}_i - \hat{\theta}_i}{\hat{\eta}_i(1 - \hat{\theta}_i)}.$$

A  $p$ -value can then easily be derived by using the spike shuffling procedure described in Section 3.1.

As a generalization of this approach one may consider to estimate the background firing rate of neuron  $i$  not only from the time bins in which at most neuron  $i$  fires, but also from those time bins, in which a maximum of  $r$ ,  $r \geq 0$ , other neurons fire ( $r = 0$  yields the case discussed above). This provides an indication whether neuron  $i$  participates in assemblies with more than  $r + 1$  neurons. However, one should be aware that for  $r > 0$  a possible participation in assemblies of smaller size (at most  $r + 1$  neurons) can obscure the participation in larger assemblies (more than  $r + 1$  neurons), because in this case we are not estimating the true background firing probability, but the firing probability resulting from background firing and participation in small assemblies.

### 3.3. Conditional Pattern Complexities (CPC)

The approach presented in Section 3.2 only considers *whether* some other neuron  $j$ ,  $j \neq i$ , fires or not, but neglects *how many* and *which* other neurons fire and *how often* they fire together with neuron  $i$ . The approaches studied in this and the next section try to exploit such additional information.

The most natural approach is to take into account *how many* other neurons fire, on average, together with neuron  $i$ . If neuron  $i$  participates in one or more (large) assemblies, there should be several time bins in which it fires together with several other neurons. Hence the average complexity of patterns involving neuron  $i$  should be larger than expected by chance. Formally, we use  $x_i = \frac{1}{k_i} \sum_{l=1}^k \tau(i \in I_l)(|I_l| - 1)$  and  $\bar{x}_i = \frac{1}{k} \sum_{l=1}^k (|I_l| - \tau(i \in I_l))$ , where  $I_l$  is the index set of the neurons that fire in the  $l$ -th time bin and  $\tau(\varphi)$  is 1 if  $\varphi$  is true and 0 otherwise. Obviously,  $x_i$  is the average pattern complexity in those time bins in which neuron  $i$  fires (with spikes of neuron  $i$  removed). On the other hand,  $\bar{x}_i$  is simply the overall average pattern complexity (again with spikes of neuron  $i$  removed). With this, a natural test statistic is

$$t_i^{\text{CPC}} = \frac{x_i - \bar{x}_i}{\bar{x}_i}.$$

A  $p$ -value can again be derived by using the spike shuffling procedure described in Section 3.1.

### 3.4. Conditional Spike Frequencies (CSF)

In a second approach we take into account *how often* other neurons fire together with neuron  $i$ . The idea is that if neuron  $i$  participates in one or more assemblies, it should fire more often together with certain other neurons (those also in the assemblies) than can be expected by chance. In order to take care of different firing rates, we use the number of excess spikes to form a test statistic: we compute for each neuron  $j$ ,  $j \neq i$ , the difference between the number of spikes observed together with a spike of neuron  $i$  and the expected number of such spikes, estimated as  $k_i \hat{\eta}_j$ . Since only excess spikes tell us about possible correlations, negative differences are ignored. Formally, the test statistic is

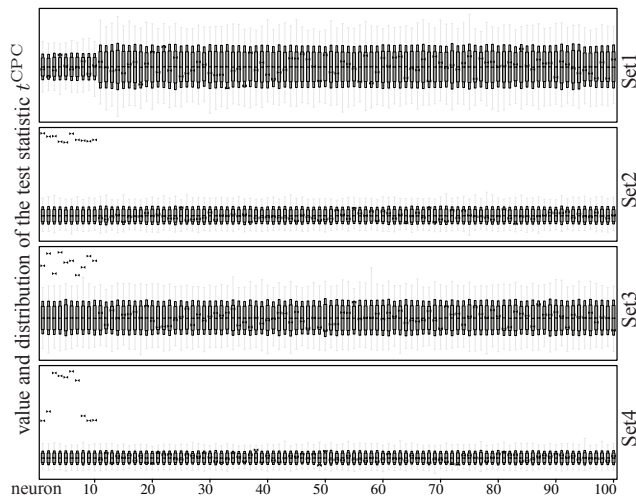
$$t_i^{\text{CSF}} = \frac{1}{n} \sum_{j=1, j \neq i}^n \tau(k_{ij} > k_i \hat{\eta}_j)(k_{ij} - k_i \hat{\eta}_j),$$

where  $k_{ij}$  is the number of time bins in which both neuron  $i$  and neuron  $j$  fire and (as above)  $\tau(\varphi)$  is 1 if  $\varphi$  is true and 0 otherwise. A  $p$ -value is derived by the same shuffling procedure described above (see Section 3.1).

### 3.5. Results

Due to limitations of space, we can present detailed results only for one of the three approaches. We chose CPC (Section 3.3) because of two reasons: in the first place, BRE (Section 3.2) proved to be a fairly weak method in our experiments, at least for  $r = 0$ . The reason is mainly the small number of time bins in which *only* the considered neuron  $i$  fires (small  $k_{i0}$ ), which leads to an unreliable estimate and thus a high variance. Performance improves for  $r = 1$  and becomes reasonably good for  $r = 2$ , but still falls short of the performance of the other two methods. Therefore we chose to discard BRE, even though we believe that the estimator of the background firing rate used in it is a relevant insight. Secondly, even though the performance of CSF (Section 3.4) was actually the best in our experiments, the approach requires an estimate of the expected number of spikes for the other neurons. This can turn out to be a severe hindrance for analyzing real spike trains, for which firing rates may be non-stationarity, and made us favor the CPC approach.

CPC results for the four data sets are depicted in Fig. 2. Each diagram shows the value of the test statistic and a box plot indicating the distribution of the spike shuffling results for each of the 100 neurons. For Set1 the test did not produce a significant result for any of the neurons, since they are independent. For the other three data sets (Set2–Set4) it is detected that the first 10 neurons have excess coincidences, thus exactly identifying the neurons involved in assemblies. For Set2, significance is highest, which corresponds to the fact



**Fig. 2.** Analysis results for Set1 to Set4 using CPC. In each panel, box plots show the distribution (thick box: 5% to 95%, thin box: 1% to 99%, gray whiskers: minimum to maximum) of the shuffling results ( $s = 10^5$  trials) for each neuron  $i$  (ids ordered, along the  $x$ -axis). Outside the whiskers, results are significant on a  $1/s$  level. The test statistic value  $t_i^{\text{CPC}}$  obtained on the actual data is shown as a bowtie.

that all correlated neurons participated with probability  $\epsilon = 1$  in the coincident events. Significance is slightly reduced for Set3, which is due to the decreased participation probability ( $\epsilon = 0.4$ ) in the coincidences. For Set4 also all neurons taking part in assemblies are reliably detected. Significance is higher for the neurons participating in both assemblies as compared to the ones participating in one of the assemblies only.

#### 4. DISCUSSION

We presented three simple test statistics to identify neurons that are involved in assemblies. All of them test for a given neuron whether it is more often involved in a coincidence spike event than can be expected by chance. To do so, BRE relies on a background rate estimator (which we believe to be relevant in its own right), CPC analyses the coincidence complexities of the parallel spike trains, and CSF aggregates pairwise frequency comparisons. To assess their performance, we applied these statistics, using a spike shuffling approach, to massively parallel spike trains generated by stochastic models that allow for defining different spike correlation structures.

Our approaches are a perfect addition to methods that provide information of *presence* of higher-order correlation in data, but do not identify the individual neurons involved. Reducing data sets to the *relevant* neurons only helps to considerably reduce computation time in further analysis steps that aim at identifying the detailed higher-order correlation structures e.g. considering information geometrical measures (e.g. [24, 25]) or approaches based on accretion [26].

**Acknowledgments:** Partially funded by the BMBF (grant 01GQ01413) and RIKEN Strategic Programs for R&D.

#### 5. REFERENCES

- [1] Abeles. *Isr. J. Med. Sci.* 18:83–92, 1982
- [2] Abeles. *Cambridge Univ. Press*, 1991
- [3] Gerstein, Bedenbaugh, Aertsen. *IEEE Trans. Biomed. Eng.* 36:4–14, 1989
- [4] Singer, Engel, Kreiter, Munk, Neuenschwander, Roelfsema. *Trends Cogn. Sci.* 1(7):252–261, 1997
- [5] Harris. *Nat. Rev. Neurosci.* 6(5):399–407, 2005
- [6] Csicsvari, Henze, Jamieson, Harris, Sirota, Barth, Wise, Buzsaki. *J. Neurophysiol.* 90:1314–1323, 2003
- [7] Brown, Kaas, Mitra. *Nat. Neurosci.*, 7(5):456–461, 2004
- [8] Aertsen, Gerstein, Habib, Palm. *J. Neurophysiol.* 61(5):900–917, 1989
- [9] Kohn and Smith. *J. Neurosci.* 25(14):3661–3673, 2005
- [10] Shmiel, Drori, Shmiel, Ben-Shaul, Nadasdy, Shemesh, Teicher, Abeles. *J. Neurophysiol.* 96(5):2645–2652, 2006
- [11] Abeles and Gerstein. *J. Neurophysiol.* 60(3):909–924, 1988
- [12] Dayhoff and Gerstein. *J. Neurophysiol.* 49(6):1334–1348, 1983
- [13] Grün, Diesmann, Aertsen. *Neural Comput.* 14(1):43–80; 81–119, 2002a,b
- [14] Schneidman, Berry, Segev, Bialek. *Nature* 440: 1007–1012, 2006
- [15] Shlens, Field, Gauthier, Grivich, Sher, Litke, Chichilnisky. *J. Neurosci.* 26(32):8254–8266, 2006
- [16] Berger, Warren, Normann, Arieli, Grün. *Neurocomput.* 70(10-12):2112–2116, 2007
- [17] Eldawlatly, Jin, Oweiss. *Neural Comput.* 21(2) (in press), 2009
- [18] Tetko and Villa. *J Neurosci. Meth.* 105:15–24, 2001
- [19] Staude, Rotter, Grün. *Soc. Neurosci. Abstr.* 103.9/AAA18, Washington, DC, 2007.
- [20] Grün, Abeles, Diesmann. *Lecture Notes Computer Sci.* 5286: 96–114, 2008
- [21] Schrader, Grün, Diesmann, Gerstein. *J. Neurophysiol.* 100(4):2165–2176, 2008
- [22] Grün, Diesmann, Grammont, Riehle, Aertsen. *J. Neurosci. Methods* 94(1):67–79, 1999
- [23] Kuhn, Aertsen, Rotter. *Neural Comput.* 1(15):67–101, 2003
- [24] Shimazaki, Amari, Brown, Grün. *this volume* 2009
- [25] Amari. *IEEE Trans. Info. Theory* 47:1701–1711, 2001
- [26] Gerstein, Perkel, Subramanian. *Brain Res.* 140:43–62, 1978