# Finding neural assemblies with frequent item set mining

**David Picado-Muiño[1]\*, Christian Borgelt[1], Denise Berger[2], George Gerstein[3] and Sonja Grün[4,5,6]**

[1] European Centre for Soft Computing, Calle Gonzalo Gutiérrez Quirós s/n, Mieres, Asturias, Spain
[2] Laboratory of Neuromotor Physiology, IRCCS Fondazione Santa Lucia, Roma, Italy
[3] Department of Neuroscience, University of Pennsylvania, Philadelphia, PA, USA
[4] Institute of Neuroscience and Medicine (INM-6) and Institute for Advanced Simulation (IAS-6), Jülich Research Centre and JARA, Jülich, Germany
[5] Theoretical Systems Neurobiology, RWTH Aachen University, Aachen, Germany
[6] Cognitive Brain Mapping, RIKEN Brain Science Institute, Wako-Shi, Japan

Cell assemblies, defined as groups of neurons exhibiting precise spike coordination, were proposed as a model of network processing in the cortex. Fortunately, in recent years considerable progress has been made in multi-electrode recordings, which enable recording massively parallel spike trains of hundred(s) of neurons simultaneously. However, due to the challenges inherent in multivariate approaches, most studies in favor of cortical cell assemblies still resorted to analyzing pairwise interactions. However, to recover the underlying correlation structures, higher-order correlations need to be identified directly. Inspired by the Accretion method proposed by Gerstein et al. (1978) we propose a new assembly detection method based on frequent item set mining (FIM). In contrast to Accretion, FIM searches effectively and without redundancy for individual spike patterns that exceed a given support threshold. We study different search methods, with which the space of potential cell assemblies may be explored, as well as different test statistics and subset conditions with which candidate assemblies may be assessed and filtered. It turns out that a core challenge of cell assembly detection is the problem of *multiple testing*, which causes a large number of *false discoveries*. Unfortunately, criteria that address individual candidate assemblies and try to assess them with statistical tests and/or subset conditions do not help much to tackle this problem. The core idea of our new method is that in order to cope with the multiple testing problem one has to shift the focus of statistical testing from specific assemblies (consisting of a specific *set* of neurons) to spike patterns of a certain size (i.e., with a certain *number* of neurons). This significantly reduces the number of necessary tests, thus alleviating the multiple testing problem. We demonstrate that our method is able to reliably suppress false discoveries, while it is still very sensitive in discovering synchronous activity. Since we exploit high-speed computational techniques from FIM for the tests, our method is also computationally efficient.

**Keywords: massively parallel spike trains, cell assembly, synchronous spike patterns, higher-order correlation, frequent item set mining, surrogate data, multi-variate significance testing**

## 1. INTRODUCTION

The principles of neural information processing are still under intense debate. Although changes in the firing rates of individual neurons are observed in relation to stimuli and behavior, the role of these changes in the joint information processing executed by *networks* of neurons is not yet clear.

As a model of network processing, cell assemblies were proposed (Hebb, 1949), which are characterized as groups of neurons exhibiting precise spike coordination. Since it can be shown theoretically that synchronous firing is most effective in generating output spikes of downstream neurons (Abeles, 1982; König et al., 1996; Schultze-Kraft et al., 2013), the *synfire chain* was proposed as a more specific model of cortical activity (Abeles, 1991). Experimental evidence from correlation analyses showed that spike synchrony indeed occurs and in particular in relation to behavior and learning (e.g., Freiwald et al., 1995; Vaadia et al., 1995; Riehle et al., 1997; Kilavik et al., 2009). However, these studies were limited to fairly small numbers of neurons recorded simultaneously, and thus it was not possible to uncover the (full) underlying correlation structure.

Fortunately, in recent years considerable progress has been made in multi-electrode recordings (e.g., Nicolelis et al., 1997; Buzsáki, 2004), which enable to record the activity of hundred(s) of neurons simultaneously. However, due to the challenges inherent in multivariate approaches (especially the combinatorial explosion of the spike patterns that need to be checked), most studies in favor of cortical cell assemblies still resorted to analyzing pairwise interactions. Although in this way the existence and functional relevance of pairwise interactions could be demonstrated in various cortical systems and behavioral paradigms (e.g., Gerstein and Aertsen, 1985; Kohn and Smith, 2005; Fujisawa et al., 2008; Feldt et al., 2009; Masud and Borisyuk, 2011), which can also be used to discover correlated groups of neurons by subsequent clustering (e.g., Berger et al., 2007; Fujisawa et al., 2008),

higher-order correlations need to be identified directly in order to recover the (full) correlation structures.

Higher-order correlations can be addressed on different levels, based on (correlation) statistics of recorded parallel spike trains and corresponding statistical tests, which focus on the following aspects: (1) test whether higher-order spike correlation is present, possibly with a lower bound on the order, but without identifying the participating neurons (e.g., Louis et al., 2010a; Staude et al., 2010a,b); (2) test for individual neurons whether they participate in synchronous spiking activity, but without identifying the groups of correlated neurons (Berger et al., 2010); (3) test for the presence of correlation as predicted by a specific correlation model (synfire chain, Abeles, 1991), that is, spatio-temporal spike patterns or propagation of synchronous spiking activity (e.g., Abeles and Gerstein, 1988; Schrader et al., 2008; Gansel and Singer, 2012; Gerstein et al., 2012); (4) actually identify the members of cell assemblies that exhibit synchronous spiking activity (e.g., Gerstein et al., 1978; Pipa et al., 2008; Feldt et al., 2009; Shimazaki et al., 2012).

In this paper we focus on the last category and solve three major challenges simultaneously: (1) detect and identify members of an active cell assembly directly as significant spike synchrony patterns (2) with an efficient and reliable statistical method that (3) is applicable to massively parallel spike trains, i.e., in the order of hundred(s) of spike trains or more.

In order to motivate and justify our new method, we study different search methods, with which the space of potential cell assemblies may be explored, and different test statistics and subset conditions with which candidate assemblies may be assessed and filtered. It turns out that a core challenge of cell assembly detection is the problem of *multiple testing*, which causes a large number of *false discoveries*. That is, many neuron groups are reported as cell assemblies that actually are not. Unfortunately, criteria that address individual candidate cell assemblies and try to assess them with statistical tests and/or subset conditions do not help much to tackle this problem.

Based on these results, we propose a new assembly detection method. The core idea of this method is that in order to cope with the multiple testing problem we shift the focus of statistical testing from specific assemblies (consisting of a specific *set* of neurons) to assemblies of a certain size (that is, with a certain *number* of neurons, thus pooling different sets of neurons). This significantly reduces the number of necessary tests, thus alleviating the multiple testing problem. Note, however, that only the focus of the statistical testing is shifted—the assemblies reported by our method are still specific sets. We demonstrate that our method is able to reliably suppress most false discoveries, while still being very sensitive in discovering synchronous activity. Since we exploit high-speed computational techniques from frequent item set mining (FIM) for the tests, our method is also computationally efficient, even if we are faced with hundred(s) of parallel spike trains.

The remainder of this paper is structured as follows: in Section 2 we introduce some notation and preliminaries on time-bin discretization and spike synchrony. In section 3 we briefly review the Accretion methodology (Gerstein et al., 1978), which forms the starting point of our investigation. In section 4 we briefly introduce FIM (Goethals, 2010; Borgelt, 2012), a data mining technique closely related to Accretion, which has a conceptually similar objective. In section 5 we compare Accretion and FIM, reveal the redundancy in Accretion's search scheme, and show how ideas from FIM can be used to eliminate it. Section 6 introduces several assembly detection criteria and how they can be incorporated into FIM approaches. In section 7 these criteria are evaluated on artificially generated data, comparing FIM and Accretion and demonstrating that the detection quality suffers severely from *false discoveries* brought about by *multiple testing*. Building on the insights gained we then introduce our novel methodology in section 8. Finally, we conclude the paper with a discussion of the merits of our method over the previously investigated approaches.

## 2. NOTIONS AND NOTATION

Throughout this paper we work with a finite set $N$ of neurons. Our raw data is a collection of $N$ (simulated) spike trains of total duration $T$, each consisting of a list of spike times in $(0, T]$.

In order to characterize and quantify synchrony among simultaneous spike trains even in the presence of temporal imprecision (regardless of whether it stems from the recording equipment or is a feature of the recorded process) we work on discretized spike data (e.g., Grün et al., 2002, "exclusive binning"). That is, we partition the time interval $(0, T]$ under consideration into time bins of equal length. Spikes (or, more precisely, spike times) corresponding to distinct neurons in $N$—say, to a subset $A \subseteq N$—are considered synchronous (i.e., a *synchronous event for A*, or simply *A-event*) if they lie in the same time bin. The amount of synchrony of a group of neurons $A \subseteq N$ (i.e., the amount of synchrony observed in the corresponding collection of spike trains) is simply the number of time bins that contain at least one *A-event* (that is, we "clip" bin entries to 1 if more than one spike of one neuron falls into a time bin).

## 3. ACCRETION

The starting point of our investigation is a statistical technique proposed in Gerstein et al. (1978), which aims at identifying *neural assemblies* (called *functional groups* in Gerstein et al. (1978) and intuitively understood as groups of neurons that tend to show *significant* synchronous spiking) in parallel spike trains. This method is *accretional* in nature, and therefore generally referred to as *Accretion* in the following: sequences of neurons are formed and extended by an iterative application of a statistical independence test between a new neuron and an already accreted sequence of neurons, based on the number of joint spiking events.

Accretion relies on Pearson's $\chi^2$ *independence test* to assess whether paired observations of two sets $A$ and $B$ of neurons, as expressed in a contingency table (see **Table 1**), are independent of each other. The counts in **Table 1** range over the number of time bins: for example, $n_{11}$ is the number of time bins that contain both an *A-event* and a *B-event*.[1] Formally, Pearson's $\chi^2$ statistic is

---

[1]Recall that an *A-event* occurs only if *every* neuron in $A$ has a spike in the time bin. That is, "not $A$" means that *not all* neurons in $A$ have a spike in the time bin and *not* that *no* neuron in $A$ has a spike in the time bin.

defined as follows, with the counts from **Table 1**, for disjoint sets of neurons $A, B \subseteq N$:

$$\chi^2(A, B) = n_{**} \frac{(n_{1*}n_{*1} - n_{**}n_{11})^2}{n_{1*}n_{0*}n_{*1}n_{*0}}.$$

Accretion considers two sets $A$ and $B$ as showing significant synchronous spiking activity according to Pearson's $\chi^2$ test if the null hypothesis of independence can be rejected with a significance level of $\alpha = 0.01 = 1\%$, that is, if $\chi^2(A, B) \geq \chi^2_{1,1-\alpha} \approx 6.635$, where $\chi^2_{1,1-\alpha}$ is the $1 - \alpha = 99\%$ quantile of the $\chi^2$ distribution with one degree of freedom. Equivalently, but often more conveniently, we may compute the $p$-value of the $\chi^2$ test, which is defined by the relation $\chi^2_{1,1-p} = \chi^2(A, B)$. In this case the test result is significant if $p \leq \alpha$.

Note, however, that Accretion considers this test result only if $n_{11}n_{**} > n_{1*}n_{*1}$, that is, if there are more synchronous spiking events than are to be expected under independence. Otherwise the two (sets of) neurons are regarded as not correlated. The difference to relying entirely on the test result is marginal, though, because of the asymmetry of the distribution.

In its first step, Accretion tests all pairs formed by singletons $A, B \subset N$ (that is, $|A| = |B| = 1$) and selects all two-element sequences that show significant synchronous spiking (according to the test criterion described above). Subsequent steps (try to) expand accreted sequences in all possible ways by adding another neuron. That is, in the $n$th step, $n < |N|$, sequences formed in the $(n-1)$th step (that is, sequences of neurons in sets $A \subset N$ with $|A| = n$) are expanded by all singletons $B \subset N \setminus A$ (that is, $|B| = 1$) and tested for independence. Those that show significant synchrony (w.r.t. Pearson's $\chi^2$ test and a chosen significance level $\alpha$) are selected and possibly expanded further in later steps. Significant sequences that cannot be expanded anymore (because no additional neuron passes a test for significant synchrony with them) are finally reported as (candidates for) neural assemblies.

Note that the original Accretion limits significant expansions to two. That is, the branching factor of the search is two: any significant sequence gives rise to at most two expanded sequences. Although it is tempting to ascribe this constraint to limitations of the computer hardware at the time (1978) and thus to omit it, it is actually still needed today due to the redundant search scheme of Accretion (see below for details). Nevertheless we ignore this branching restriction for the following considerations in order to avoid some technical, but largely irrelevant complications. That is, we assume for now that any additional neuron that passes a synchrony test with an accreted sequence gives rise to an expanded sequence.

Without a branching restriction, we can formalize the implicit characterization of a neural assembly underlying Accretion in terms of *subset conditions*. We say that $A \subseteq N$ constitutes a *synchronous group* if it satisfies the following:

- If $|A| = 2$ then the two singleton subsets of $A$ must show significant synchrony (as evaluated by Pearson's $\chi^2$ independence test w.r.t. a significance level $\alpha$, see above).
- If $|A| > 2$ then there has to exist a subset $B \subset A$ with $|B| = |A| - 1$ that is a *synchronous group* and that shows significant synchrony with the remaining neuron in $A$, that is, with $A \setminus B$ (again as evaluated by Pearson's $\chi^2$ test). (Note the recursive structure of this definition.)

Accretion reports as neural assemblies only significant sequences that cannot be expanded anymore. That is, it never reports proper prefixes (and thus subsets) of a detected assembly. Hence we may say that Accretion regards as neural assemblies *maximal synchronous groups*. Here "maximal" expresses that there does not exist a superset that is also a synchronous group.

As an example, **Figure 1A** shows an extremely simple binned parallel spike train for four neurons $a$, $b$, $c$, and $d$ (that is, $N = \{a, b, c, d\}$) and 10 time bins. For each time bin the neurons having a spike in it are marked. **Figure 1B** shows how this data is processed by Accretion in three steps, using a significance level $\alpha = 0.2$. (Clearly, due to the very low number of time bins, no significant results could be obtained for $\alpha = 0.01$.) As a result, Accretion reports the four sequences $abcd$, $bacd$, $cd$ and $dc$, which are the maximal synchronous groups.
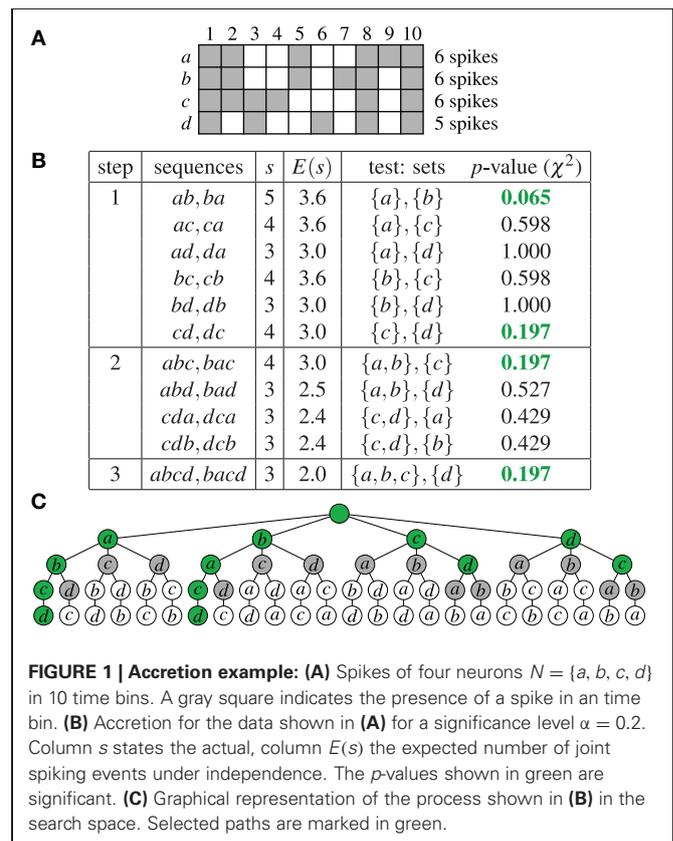


**FIGURE 1 | Accretion example: (A)** Spikes of four neurons $N = \{a, b, c, d\}$ in 10 time bins. A gray square indicates the presence of a spike in an time bin. **(B)** Accretion for the data shown in **(A)** for a significance level $\alpha = 0.2$. Column $s$ states the actual, column $E(s)$ the expected number of joint spiking events under independence. The $p$-values shown in green are significant. **(C)** Graphical representation of the process shown in **(B)** in the search space. Selected paths are marked in green.

**Table 1 | A 2 × 2 contingency table for binary events A and B.**

|  | 0 (not B) | 1 (B) | Sum |
|---|---|---|---|
| **0 (not A)** | $n_{00}$ | $n_{01}$ | $n_{0*}$ |
| **1 (A)** | $n_{10}$ | $n_{11}$ | $n_{1*}$ |
| **sum** | $n_{*0}$ | $n_{*1}$ | $n_{**}$ |

As a further illustration **Figure 1C** shows the complete search space of the Accretion method for this example, that is, all sequences that are potentially explored. The green nodes correspond to significant test results (with the empty sequence—root node—and the singletons being considered significant by default). Gray nodes are visited, but do not yield significant test results. White nodes are not visited. Accretion reports the sequences corresponding to the deepest green nodes, listing all neurons on the path from the root to these nodes.

## 4. FREQUENT ITEM SET MINING

FIM was originally motivated by the desire to find regularities in the shopping behavior of customers (of supermarkets, mail-order companies, on-line shops etc.) by identifying sets of products that are frequently bought together [*market basket analysis*, see the seminal paper (Agrawal et al., 1993) and the surveys (Goethals, 2010; Borgelt, 2012)]. Conceptually, this task is obviously the same as finding sets of neurons that (frequently) fire together in parallel spike trains, which establishes the relevance of FIM for detecting neural assemblies in parallel spike trains.

Formally, we consider an item base $I = \{i_1, \ldots, i_n\}$, $n \in \mathbb{N}$, and a database $\mathcal{T} = \{t_1, \ldots, t_m\}$ of transactions, $m \in \mathbb{N}$, where $t_k$ (the $k$th transaction) is a pair $\langle k, J_k \rangle$ consisting of a unique identifier $k$ and a subset of items $J_k \subseteq I$. In our context the item base $I$ is the set $N$ of neurons and the transactions are determined by the time bins into which we partition the recording time interval $(0, T]$: the set of neurons with spike times lying in the $k$th time bin constitutes $J_k$ in the $k$th transaction $t_k = \langle k, J_k \rangle \in \mathcal{T}$. **Table 2** summarizes the correspondences.

**Table 2 | Meaning of basic notions of frequent item set mining.**

| Mathematics | Market basket analysis | Spike train analysis |
|---|---|---|
| Item | Product | Neuron |
| Item base | Set of products | Set of neurons |
| Transaction *id* | Customer | Time bin |
| Transaction | Set of products bought by a customer | Set of neurons firing in a time bin |
| Frequent item set | Set of products frequently bought together | Set of neurons frequently firing together |

A transaction $t = \langle k, J_k \rangle \in \mathcal{T}$ is said to support a subset $J \subseteq I$ if $J \subseteq J_k$. The number of transactions in $\mathcal{T}$ that support $J$ is called the *support* of $J$ in $\mathcal{T}$ and is denoted by $s_{\mathcal{T}}(J)$ (or simply by $s(J)$ whenever $\mathcal{T}$ is clear from the context). Frequent item sets are defined based on some user-specified threshold $s_{min}$: $J$ is called *frequent* in $\mathcal{T}$ if $s_{\mathcal{T}}(J) \geq s_{min}$.

The FIM problem consists in finding all subsets of $I$ that are frequent (in our context: all sets of neurons $A \subseteq N$ that show frequent synchronous emission of spikes, that is, a number of spike-time coincidences at least $s_{min}$). The search for all such item sets exploits that the support operator $s$ is *anti-monotone*: for $J_1, J_2 \subseteq I$ and $J_1 \subseteq J_2$ we have $s(J_2) \leq s(J_1)$. As a consequence, if $s(J_1) < s_{min}$ we also know $s(J_2) < s_{min}$. In words: *no superset of an infrequent item set can be frequent*. This statement is also known as the *Apriori property*.

The search space is $\mathcal{P}(I)$, the power set of $I$ (that is, the collection of all subsets of $I$). $\mathcal{P}(I)$ together with the subset relations between its elements is a partially ordered set, which is conveniently represented as a Hasse diagram (see **Figure 2A**). The search through $\mathcal{P}(I)$ is made irredundant by assigning a unique parent to each item set, which turns the search space into a tree (see **Figures 2B,C** for two variants). In such a search tree every item set can be reached on exactly one path and therefore it is visited *at most once* in the search. Details about efficient algorithms and data structures to actually carry out the search can be found, for example, in Goethals (2010) and Borgelt (2012).

As an illustration, **Figure 3** shows three data sets with distinct neural assemblies. **Figure 4** shows, in the search space structured as a tree according to **Figure 2B**, the frequent item sets that can be found in these data sets for $s_{min} = 3$ (blue and red nodes). If, in analogy to Accretion, only the *maximal frequent item sets* are reported (i.e., no superset is frequent), FIM yields the sets in the nodes marked in red. Note that the search can be pruned with the Apriori property (that is, no supersets of infrequent item sets are explored) without affecting the result: all frequent item sets can still be reached from the root.

## 5. AVOIDING REDUNDANT SEARCH

As explained above, Accretion derives sequences of neuron ids composing spike patterns rather than sets, which we are actually interested in. As a consequence, it suffers from considerable
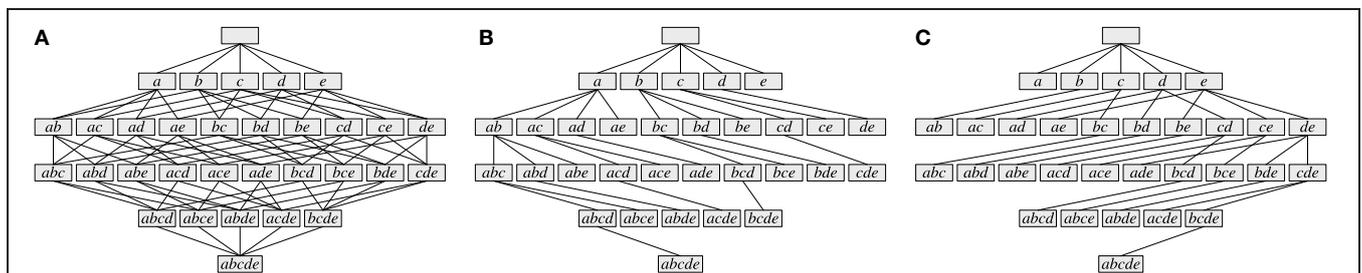


**FIGURE 2 | The search space for five neurons $N = \{a, b, c, d, e\}$.**
**(A)** Hasse diagram: a graph in which each possible set is a node and any two sets $I, J \subseteq N$ with $I \subset J$ are connected by an edge if and only if $\nexists K : I \subset K \subset J$. **(B)** A selection of edges from the Hasse diagram that reduces it to a search tree. Parents are assigned based on the alphabetical order of the neuron identifiers according to parent$(I) = I \setminus \{\max(I)\}$. **(C)** An alternative selection of edges from the Hasse diagram that also reduces it to a search tree. Parents are assigned based on the alphabetical order of the neuron identifiers according to parent$(I) = I \setminus \{\min(I)\}$.

redundancy. This is demonstrated in **Figure 5A**, which shows the search carried out for a data set with four neurons if all tests yield significant results. The same set of all four neurons is considered $4! = 24$ times (since the search tree has 24 leaves). Even worse, the same statistical test is executed multiple times: leaves having the same color correspond to the same test (the leaf neuron is tested against the *set* of neurons on the path leading to it).

Clearly, this redundancy stems from the fact that the Accretion test ignores the order of the already accreted neurons for the test. Note that it is the main reason why Accretion still needs a branching restriction even with modern computer hardware: without it, an assembly with $n$ neurons is considered $n!$ times, which becomes infeasible already for moderately large $n$.

However, even if we ensure that each test is executed only once, the same set of $n$ neurons is still considered $n$ times (four times in **Figure 5A**, corresponding to the four colors). Although the tests differ, it would suffice to consider each set only once (executing $n$ tests on it if desired). In contrast to this, FIM works on sets and with the search space structured as a tree (see **Figure 2B** or **Figure 2C**), it guarantees that each set is visited *at most once* (no redundancy). The potential improvements are illustrated in **Figure 5B**, which show the search space of Accretion for four neurons. Marked in red is the part that is (potentially) explored by FIM (cf. also **Figure 2B**).

## 6. CRITERIA FOR ASSEMBLY DETECTION

FIM prunes the search and selects results based only on the support of item sets. As a consequence, it may produce results that are not considered by Accretion. For example, a set of neurons may be frequent simply because the member neurons have many spikes and thus are likely to exhibit (non-significant) synchronous spiking. The statistical tests in Accretion eliminate such cases. However, various forms of *statistical testing* can fairly easily be added to FIM as a further evaluation criterion, because the support values of different item sets are all that is needed to compute the test statistics.

In addition, Accretion requires a path of significant test results, which we described above in terms of *subset conditions*. However, such subset conditions may also be added to FIM and we study several variants below.

### 6.1. STATISTICAL TESTING

Accretion's sequence-based scheme fixes the test to be carried out: the neuron added last is tested against the set of already accreted neurons. In FIM, since each set is considered at most once, we have a choice of $n$ tests, where $n$ is the number of neurons in the set: each neuron may be tested against the $n - 1$ other neurons. Note that Accretion carries out these tests in different branches of the search tree. Note also that in FIM's search tree (cf. **Figure 2B** or **Figure 2C**) there is also a neuron added last, but this neuron depends on an essentially arbitrary global order of the neurons and thus should not determine the test.

Most naturally, we should use an *aggregate* of the results of the $n$ tests to evaluate a set of $n$ neurons. Following the
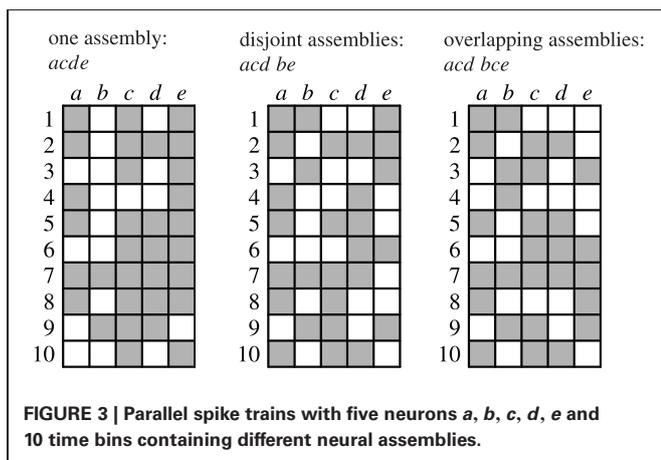


**FIGURE 3 | Parallel spike trains with five neurons *a*, *b*, *c*, *d*, *e* and 10 time bins containing different neural assemblies.**
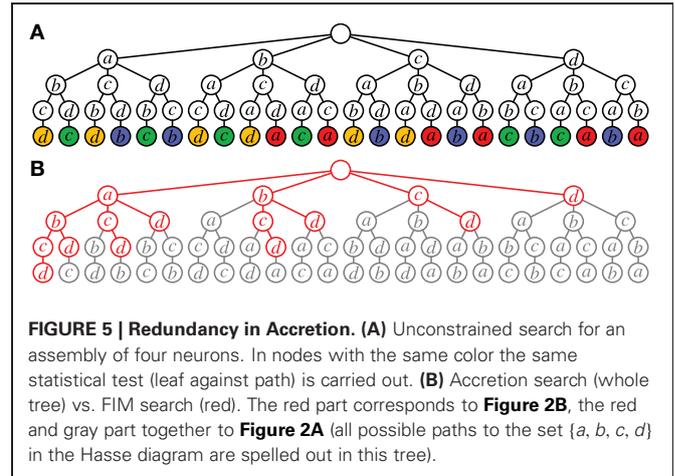


**FIGURE 5 | Redundancy in Accretion. (A)** Unconstrained search for an assembly of four neurons. In nodes with the same color the same statistical test (leaf against path) is carried out. **(B)** Accretion search (whole tree) vs. FIM search (red). The red part corresponds to **Figure 2B**, the red and gray part together to **Figure 2A** (all possible paths to the set {*a*, *b*, *c*, *d*} in the Hasse diagram are spelled out in this tree).
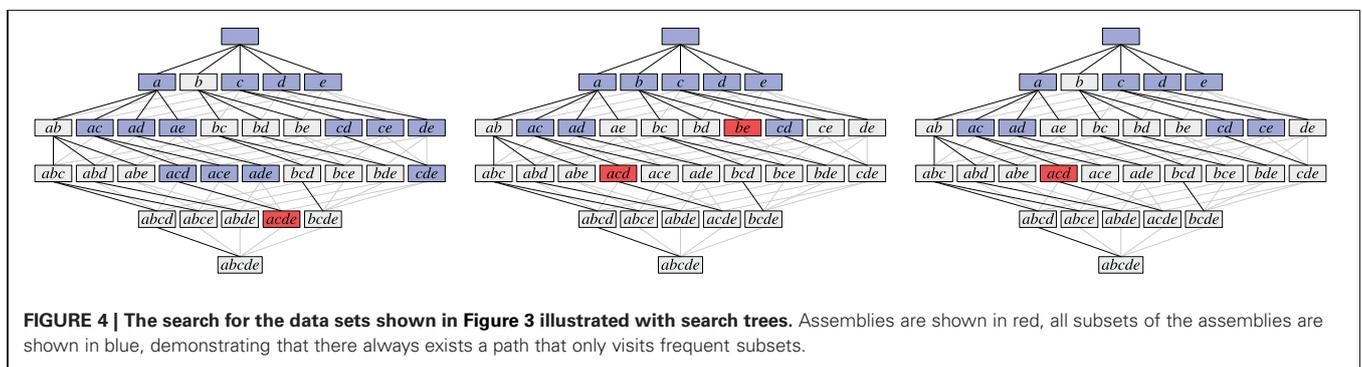


**FIGURE 4 | The search for the data sets shown in Figure 3 illustrated with search trees.** Assemblies are shown in red, all subsets of the assemblies are shown in blue, demonstrating that there always exists a path that only visits frequent subsets.

general principle of statistics to consider the *worst case*, it is most appropriate to use the *maximum* of the *p*-values of the *n* tests. That is, a set of *n* neurons is judged to exhibit significant synchrony if the largest *p*-value of the *n* tests does not exceed the chosen significance level $\alpha$. In other words, the "most independent" neuron determines the evaluation of the set. Note that Accretion effectively evaluates a set rather by the "least independent" neuron, because a set of neurons is already reported if *one* of the search tree paths leading to it yields a significant test result. That is, Accretion uses the *minimum* of the *p*-values, thus considering, in a certain sense, the *best case*. Note also that Fisher's method of combining *p*-values bearing on the same overall hypothesis (Fisher, 1925) is an alternative to taking the maximum.

Apart from this adapted test procedure, we may ask whether Pearson's $\chi^2$ independence test is the most appropriate choice. This test assumes that the (discrete) probability of observed frequencies (in the contingency table) can be approximated by the (continuous) $\chi^2$ distribution. Due to the approximation, some error is introduced, which may lead to incorrect test decisions. The approximation error is generally the larger, the lower the number of degrees of freedom, which is only one for our case of a $2 \times 2$ contingency table. In addition, for such a table an acceptable approximation requires that all expected counts are at least 5. This renders the test not very well suited for rare events like spike coincidences. As a consequence, it is worthwhile to look for alternatives, which include:

- **Yates's correction for continuity** (or Yates's $\chi^2$ test).
  Replace the $\chi^2$ value of Pearson's test with

$$\chi^2_{\text{Yates}}(A, B) = n_{**} \frac{(|n_{1*}n_{*1} - n_{**}n_{11}| - 0.5n_{**})^2}{n_{1*}n_{0*}n_{*1}n_{*0}}.$$

  This correction increases the *p*-value and thus prevents an overestimation of significance, especially if the contingency table has a cell with an expected count less than 5.

- **G-statistic** (or *G*-test).
  A likelihood-ratio based statistical significance test that replaces the $\chi^2$ value of Pearson's test with

$$G(A, B) = 2n_{**} \sum_{i=0}^{1} \sum_{j=0}^{1} \ln \frac{n_{ij}}{n_{i*}n_{*j}}.$$

  The *G*-statistic achieves a better approximation to the theoretical $\chi^2$ distribution, especially if $n_{11}n_{**} > 2n_{1*}n_{*1}$.

- **Fisher's exact test.**
  Compute the *p*-value by summing over the (exact) probabilities of contingency tables with the same marginals that are at least as extreme as the actual table. Although there are alternatives, we rely on the most common choice that "at least as extreme as" means "at most as probable as." This test avoids approximation, but is costly to compute.

Note that for both Yates's test and the *G*-test the test decision is made (as for Pearson's test) with a $\chi^2$ distribution with one

degree of freedom. That is, the *p*-value is defined by the relations $\chi^2_{1,1-p} = \chi^2_{\text{Yates}}(A, B)$ and $\chi^2_{1,1-p} = G(A, B)$, respectively.

## 6.2. SUBSET CONDITIONS

Since support is anti-monotone (and thus the Apriori property holds), FIM guarantees that all sets of neurons exhibiting at least $s_{\min}$ spike coincidences are explored. Accretion's search, however, is guided by statistical test results (or the underlying *p*-values), which do *not* have this property: if a set of neurons does not exhibit significant synchronous spiking, we have no guarantee that there are no supersets that are significant. The *p*-value of an independence test on a superset may be higher or lower than the *p*-value of a test on the set (that is, *p*-values are neither monotone nor anti-monotone). Nevertheless, Accretion does not explore any such supersets, which can reduce the results. Above (in section 3), we formalized this behavior by *subset conditions* that a set of neurons has to satisfy in order to be regarded a *synchronous group*.

Since the FIM search does not impose any such subset conditions, we are free to explore alternatives:

- **No subset conditions.** A set $A \subseteq N$ of neurons is a synchronous group if all subsets $B \subset A$ with $|B| = |A| - 1$ show significant synchrony with the remaining neuron in $A$, that is, with $A \setminus B$. In other words, $A$ passes the statistical test described above (and, of course, $s(A) \geq s_{\min}$).

- **Weak subset conditions.** A set $A \subseteq N$ of neurons is a synchronous group if it satisfies the condition of the first point and, for a user-specified minimum set size $r$, satisfies either $|A| \leq r$ or, if $|A| > r$, that *at least one* subset $B \subset A$ of cardinality $|B| = |A| - 1$ is a synchronous group.

- **Strong subset conditions.** A set $A \subseteq N$ of neurons is a synchronous group if it satisfies the condition of the first point and, for a user-specified minimum set size $r$, satisfies either $|A| \leq r$ or, if $|A| > r$, that *all* subsets $B \subset A$ of cardinality $|B| = |A| - 1$ are synchronous groups.

**Figures 6A,B** illustrate weak subset conditions in the FIM search space for the set $\{a, c, d, e\}$ (in red) for $r = 2$ and $r = 3$, respectively. In **Figure 6A** we have a path from $\{a, c, d, e\}$ down to the pair $\{c, d\}$ (dark blue) and, in **Figure 6B**, down to the triplet $\{a, c, d\}$ (dark blue). Weak subset conditions mean that there *exist* such paths containing only synchronous groups.

**Figures 6C,D** illustrate strong subset conditions in the FIM search space for the set $\{a, c, d, e\}$ (in red) for $r = 2$ and $r = 3$, respectively. In **Figure 6C** we have paths from the set $\{a, c, d, e\}$ down to all pairs contained in it (dark blue) and, in **Figure 6D**, down to all triplets contained in it (dark blue). Strong subset conditions mean that *all* such paths contain only synchronous groups (whereas weak only asks for at least one path).

Note that weak subset conditions, for $r = 2$, are in fact very similar to Accretion's subset conditions, because Accretion explores a certain sequence only if all of its prefixes produced significant test results. These prefixes form the path required by weak subset conditions. The difference is, of course, that Accretion only requires the one test with the last added neuron to be significant,

**FIGURE 6 | Illustration of different types of subset conditions (cf. Figure 4A). (A)** Weak subset conditions down to pairs; **(B)** Weak subset conditions down to triplets; **(C)** Strong subset conditions down to pairs; **(D)** Strong subset conditions down to triplets.
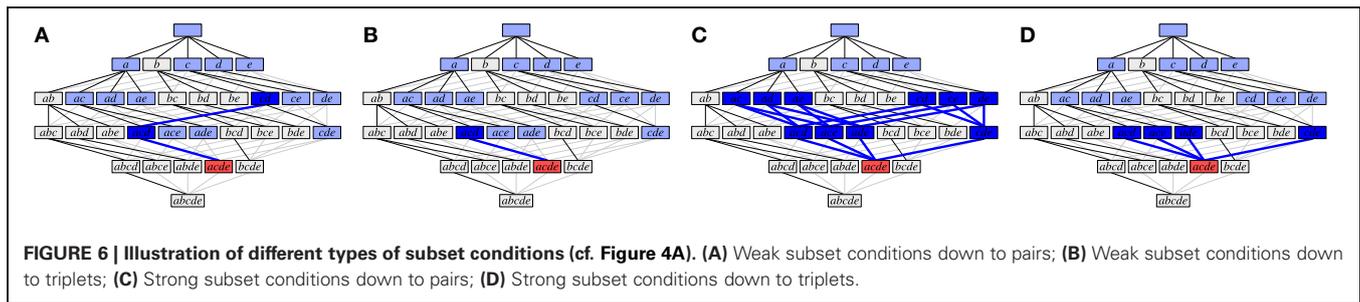
while we require of a synchronous group that *all* tests of single neurons against all others are significant.

### 6.3. MAXIMAL AND CLOSED SYNCHRONOUS GROUPS

Following Accretion, we may report as (candidate) neural assemblies the *maximal synchronous groups*, that is, those synchronous groups for which no superset is a synchronous group. However, such a choice may lose valuable information: if the data contains a significant number of synchronous spiking events of a neural assembly $A$, it is not unlikely that by chance a neuron $a \notin A$ produces a spike at a few of these events. This may render the set $A \cup \{a\}$ a synchronous group (if the number of accidentally synchronous spikes is large enough, for which 2 or 3 spikes may already suffice—see experiments below). In such a case, the subset $A$ exhibits (many) more synchronous spiking events, which are not considered if only $A \cup \{a\}$ is reported as the maximal synchronous group.

This problem can be addressed by drawing on a notion that is well known in FIM, namely so-called *closed frequent item sets*. A frequent item set is called *closed* if no superset has the same support (while it is *maximal* if no superset is frequent). Since it is unlikely that accidentally synchronous spikes of a neuron $a \notin A$ occur together with *all* synchronous spiking events of an assembly $A$, a restriction to closed sets still reports the actual assembly $A$. Therefore, we may report *closed synchronous groups*, which are synchronous groups no superset of which is a synchronous group with the same support. Note that we do not lose anything in this way, because all *maximal synchronous groups* are obviously closed.

Note also that closed synchronous groups avoid certain unintuitive effects of changing the minimum support $s_{min}$: while increasing the minimum support may render certain sets maximal which were not maximal before (as supersets may be suppressed), it only eliminates closed sets with a lower support.

### 7. EXPERIMENTAL RESULTS

We report results for our assembly detection methods on two types of data: independent parallel spike trains (to check for *false positives*) and parallel spike trains with correlated subsets of neurons (to check for *false negatives*):

- *Independent spike trains.* 1000 trials, each with 100 parallel spike trains ($|N| = 100$), were generated independently as Poisson processes with a constant rate of 20 Hz. The duration of all trials was 3 s and the length of the time bins was 3 ms (1000 time bins).

- *Correlated spike trains.* Trials that contain potential assemblies were generated by injecting synchronous spikes for a subset of the neurons into 100 parallel spike trains, which were independently generated as Poisson processes with a constant rate of 20 Hz for the neurons outside the subset and a constant rate of 20 Hz minus the rate of the synchronous spikes for the neurons in the selected subset. The size $z$ of the neuron subset was varied from 2 to 9, the number $c$ of injected synchronous spiking events also from 2 to 9, and thus a total of 64 pairs $\langle z, c \rangle$ were tested. For each pair 1000 trials were generated (3 s duration, 3 ms time bin length, i.e., 1000 time bins), resulting in a total of 64,000 data sets with injected assemblies.

On these data sets we compared Accretion and different FIM-based approaches, namely with no subset conditions as well as weak and strong subset conditions down to pairs (that is, $r = 2$). For each approach we tried all of the statistical tests considered above: Pearson's $\chi^2$-test, Yates's test, $G$-test and Fisher's exact test. In addition, we executed FIM without any statistical test (and thus without any subset conditions). That is, the minimum support $s_{min}$ was the only selection criterion, for which $s_{min} = 2$ was generally chosen (including the methods with subset conditions and statistical tests).

**Figure 7** shows the results on independent spike trains in the form of pattern spectra [like those used for spatio-temporal spike patterns in Gerstein et al. (2012)]. Each bar chart refers to a detection method (Accretion or FIM with different subset conditions; row of the chart grid) and a test statistic (column) and shows the decimal logarithm of the average number of detected patterns (in the sense of *maximal synchronous groups*) subdivided according to the size $z$ of the group of neurons underlying the pattern (number of contained neurons) and the number $c$ of coincident spiking events (number of time bins with a spike from all neurons in the group). For comparisons, a further bar chart at the bottom shows the result of applying FIM without any subset conditions nor statistical testing (that is, simply an average count of all maximal frequent item sets).

Since the data was generated as independent Poisson processes, all detected patterns are clearly *false discoveries* or *false positives*. The number of such false discoveries is fairly high (note that due to the logarithmic scale, 1, 2, and 3 on the vertical axis stand for 10, 100, and 1000 patterns, respectively, *per trial*). Even fairly pronounced cases like three neurons with five coincident spiking events or five neurons with three coincident spiking events

**FIGURE 7 | Independent spike trains:** decimal logarithm of the average number of (significant) patterns found in 1000 trials (white squares: zero patterns, gray bars: higher than chart). Each row of the chart grid corresponds to a detection method (diagram titles, top line; "None", "Weak 2" and "Strong 2" refer to subset conditions used with FIM, $r = 2$), each column to a statistical test (diagram titles, bottom line). The diagram at the left bottom shows the result of pure FIM (no subset conditions, no statistical test, $s_{min} = 2$).

occur in several of the trials. The alternative test statistics (Yates, $G$-test and Fisher instead of Pearson's $\chi^2$) reduce the number of false discoveries (as expected, because they are less prone to overestimating significance), but are far from solving the problem, especially, since they are effective mainly for small patterns with few coincidences. Subset conditions have a similar effect:

weak subset conditions achieve essentially the same result as original Accretion (as expected, see above) and thus suppress some of the false discoveries made without them. However, only strong subset conditions are able to bring the false discoveries down to an acceptable level (assuming that pair patterns are generally ignored).

Unfortunately, strong subset conditions are no solution either, because they almost prevent discoveries altogether, even correct ones. This can be seen in **Figure 8**, which shows the results on correlated spike trains as the rate (computed over 1000 trials) of *false negatives* for all detection methods, again subdivided according to the group size $z$ and the number $c$ of (injected) coincidences. (Note that the actual number of coincidences in the data may be higher, because the injected coincidences may be supplemented by accidental coincidences, but also lower, due to "clipping" caused by the time binning.) An injected pattern
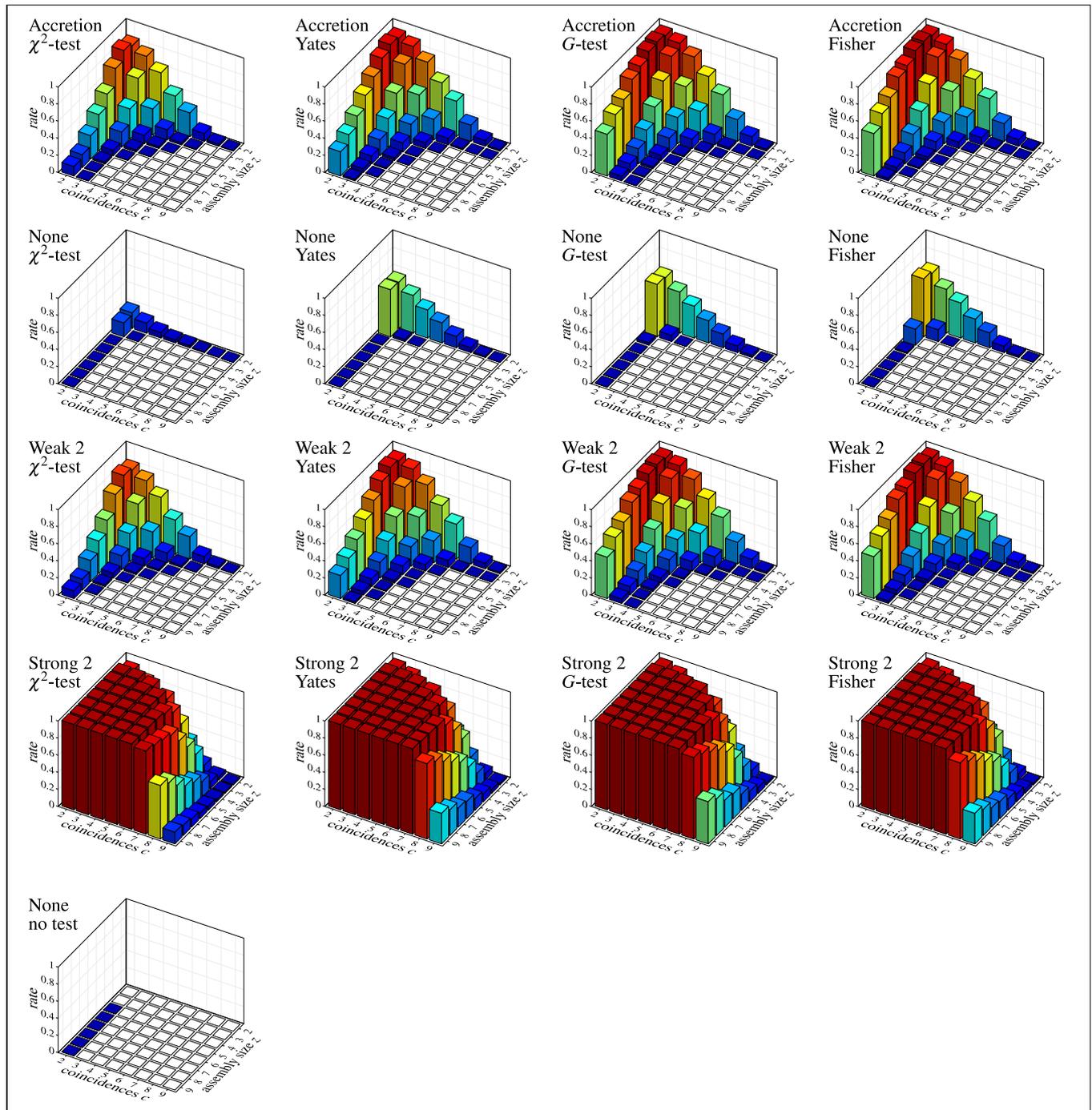


**FIGURE 8 | Correlated spike trains:** false negative rates over 1000 trials, each with a specific injected pattern, characterized by an assembly size $z$ and a number $c$ of injected coincidences. Rows and columns of the chart grid as in **Figure 7**. An injected neural pattern is counted as *not* detected (*false negative*)

if it is *not* contained in any of the *maximal synchronous groups* produced by the used detection method. Pure FIM (see diagram on the left) necessarily always detects all injected patterns perfectly, unless they are reduced to less than two coincidences by "clipping" caused by the time binning.

is counted as a *false negative* if it is not contained in any of the *maximal synchronous groups* reported by a method. That is, it *is* counted as detected even if it is not exactly among the reported patterns, but only a superset (with additional neurons) has been discovered.

Despite this somewhat lax criterion, FIM with strong subset conditions is hardly able to detect the injected patterns, unless a not too large assembly (fairly few neurons) exhibits a fairly large number of coincidences. Even clear cases like eight neurons firing together seven times are not detected. The reason, however, is obvious: with strong subset conditions and $r = 2$, all pairs contained in a group of neurons have to test positive in order for the group to qualify as a pattern. The more neurons there are in a group, the more pairs there are and thus the more likely it is that one of them is, by accident, not significant, unless the number of (injected) coincidences is fairly large. The situation is improved with $r = 3$, but the detection rate still suffers for larger assemblies (see **Figure 9A**), and the more so if we consider larger time bins (**Figure 9B**). Furthermore, higher firing rates worsen the situation as well (**Figure 9C**).

Accretion and FIM with weak subset conditions exhibit better detection capabilities, but are far from impressive either. Only if we abandon subset conditions, the detection rate is significantly improved (few false negatives). Note that FIM without subset conditions and relying only on minimum support (no statistical tests) necessarily detects all injected patterns, unless the coincident spiking events are reduced to less than two by "clipping" due to the time binning.

In general, the tougher the subset conditions of the assembly detection method (including a lower value of $r$) the lower the number of false positives, but the higher the rates of false negatives. Alternative statistical tests have a similar effect: Yates's test, $G$-test and Fisher's exact test reduce the number of false positives, but at the price of higher rates of false negatives. The same holds for the significance level (not shown, but obvious): the lower $\alpha$, the fewer false discoveries are made, but at the price of fewer correct discoveries (more false negatives). If we take into account what false positives are suppressed (by whatever approach), namely mainly small patterns with few coincidences (back corners of the bar charts), while larger patterns with more coincidences are essentially unaffected, one may wonder whether statistical tests and subset conditions are actually worth the effort (because both increase the computation time considerably compared to a pure FIM approach).

Generally we can say that the core problem of a reliable detection of neural assemblies is the large number of false positives, which the discussed methods cannot reduce without severely harming the detection sensitivity of the method. This number of false discoveries may appear to be surprising at first sight, because we used a significance level of $\alpha = 1\%$. However, this applies only to an *individual test*, whereas we are executing a huge number of tests in the search. That is, we face the problem of *multiple testing*, due to which we lose an effective control over the significance level. In simple terms: if we execute 1000 (independent) tests with $\alpha = 1\%$, we should expect about $1000 \cdot 1\% = 10$ positive test results, simply as chance events, signifying nothing. In



**FIGURE 9 | Selected other false positive and false negative results,** demonstrating the effect of the parameter $r$ (that is, the lower size limit for the subset conditions), the effect of the time bin width and the firing rate in combination with strong subset conditions down to triplets, and the effect of using closed instead of maximal sets or synchronous groups. Note the logarithmic scale (decimal logarithm) of the vertical axis of the false positive bar charts.

our search, however, we even execute millions of tests. For example, 100 recorded neurons allow for $\binom{100}{3} = 161,700$ triplets and $\binom{100}{4} = 3,921,225$ quadruplets. As a consequence, even though it is very unlikely that, say, four *specific* neurons fire together three times if they are independent (such an event has a *p*-value of less than $10^{-6}$ in our experimental setup with the testing methodology described above), it is fairly likely that we observe *some set* of four neurons firing together three times. Indeed, in our experiments we see, on average, more than one such pattern per trial.

Unfortunately, because of the excessive number of tests executed in the search, standard methods to handle the multiple testing problem (the like *Bonferroni correction* (Bonferroni, 1935; Abdi, 2007), the *Holm-Bonferroni method* (Holm, 1979), or the *Benjamini-Hochberg procedure* Benjamini and Hochberg (1995); see Dudoit and van der Laan (2008) for an overview) require *p*-values so low that they are extremely unlikely to be obtained from actual data, and also lead to a very high rate of false negatives: effectively we cannot expect to make *any* discoveries anymore. As a consequence, we have to change the testing methodology in order to obtain a suitable method (see the next section).

Finally, the diagrams in **Figure 7** show that a detection based on *maximal synchronous groups* causes strange effects: while pure FIM reports no pairs with 9 coincidences and only few with 8, all other methods detect several such pairs (or at least more than pure FIM). The reason is that these patterns are, of course, present in the data, but they are not *maximal frequent item sets*. These patterns rather have supersets with three or four neurons (exhibiting fewer coincidences), which are reported by pure FIM and thus eliminate the pairs. These supersets are suppressed, though, by the statistical tests or the subset conditions of the other detection methods, rendering the pairs maximal. Note that these patterns are found with pure FIM if we report the *closed synchronous groups* (closed frequent item sets) as demonstrated in **Figure 9E** as compared to **Figure 9D** (which repeats the bottom left bar chart of **Figure 7**). Note that the pattern counts are affected the less (by reporting closed patterns instead of maximal) the larger the patterns, even if a statistical test is used (see **Figure 9F** for an example).

## 8. ASSEMBLY DETECTION WITH FIM

Based on the insights gained in the previous section, we propose now an assembly detection method based on (pure) FIM that reduces the problem of multiple testing considerably. In this method we no longer look at specific neuron groups, which is the main reason for the huge number of tests. Rather we rely on the rationale that a pattern of a certain size $z$ and exhibiting a certain number $c$ of coincident spiking events cannot reliably identify an assembly if a counterpart—that is, a neuron group with the same size $z$ (but possibly different composition) and the same number $c$ of coincidences—occurs in properly generated surrogate data (rendering the neurons independent). The reason is that the occurrence of a counterpart in surrogate data demonstrates that the pattern could be a chance event and thus that it is not significant. To be more specific, our assembly detection method works as follows:

- *FIM on Original Data*. We apply FIM to our original (binned) data and report all *closed frequent sets* of neurons together with their support, where a neuron set is called *closed* if all of its supersets have a lower support. We prefer closed over maximal sets due to the reasons pointed out above, especially the loss of support information incurred by reporting only maximal sets. We recommend $s_{min} = 2$, but higher values may also be used.
- *FIM on Surrogate Data*. In order to determine which closed frequent neuron sets found in the original data may be due to chance events, we generate surrogate data sets. That is, we create modifications of the original data that are intended to keep all of their essential features except spike synchrony—the feature we are testing for. [For a survey and analysis of the methodology to generate surrogate data from parallel spike trains, see, for example, (Grün, 2009; Louis et al., 2010b,c)]. To each surrogate data set we apply FIM and collect the *closed frequent sets* together with their support. More specifically, we collect the signatures $\langle z, c \rangle$ (size $z$ and coincidences $c$) of found patterns. Afterward we eliminate from the closed frequent neuron sets found in the original data all sets for which a counterpart (that is, same size $z$ and same number $c$ of coincidences, but possibly different composition, that is, different underlying set of neurons) was found in a surrogate data set, since such sets could be chance events (see below for a more detailed justification and discussion of this procedure).

Note that this procedure still suffers from a certain amount of *multiple testing*: every pair $\langle z, c \rangle$ that is found by FIM in the original data gives rise to one test. However, the pairs $\langle z, c \rangle$ are *much fewer* than the specific neuron sets that are considered in all methods discussed above. As a consequence, simple approaches like *Bonferroni correction* (Bonferroni, 1935; Abdi, 2007) become feasible. That is, we divide the desired overall significance level α by the number $n$ of tests to obtain the significance level $\alpha' = \alpha/n$ for each individual test. Since in practice we can expect to find only a few dozen pairs $\langle z, c \rangle$ in the data to analyze, we obtain significance levels $\alpha'$ that leave us good chances of making detections. The number $n$ of tests (that is, $\langle z, c \rangle$ pairs) may even be reduced further by the insight that patterns with signatures like $\langle 2, 2 \rangle$, $\langle 3, 2 \rangle$, $\langle 2, 3 \rangle$ etc. are certainly discovered in the data, but we do not consider these patterns as candidates for assemblies right from the start. Only pairs $\langle z, c \rangle$ with sufficiently large $z$ and/or $c$ need to be counted, for which we are actually willing to accept the underlying neuron sets as assemblies.

Technically, the significance level α enters the testing procedure as the number of surrogate data sets to be generated. For example, if we choose $\alpha = 1\%$ and there are $n = 30$ pairs $\langle z, c \rangle$ in the original data which we actually want to submit to a test, we have to generate $k = n/\alpha = 3000$ surrogate data sets. The reason is that of the results on the original data we keep only such neuron sets for which we do *not* see a counterpart in any of the surrogate data sets. That is, if we do not observe any occurrence of a specific pair $\langle z, c \rangle$ in $k = 3000$ surrogate data sets, then we can estimate the probability of this pair $\langle z, c \rangle$ occurring by accident as less than $p = 1/k = 1/3000$. This is the *p*-value of the test and since it is $p \leq \alpha' = \alpha/n = 0.01/30 = 1/3000$, any pattern with the signature $\langle z, c \rangle$ is significant.

Note that the resulting procedure is computationally feasible due to the sophisticated high-speed implementations that are available for closed FIM.[2] Even running FIM on thousands of surrogate data sets takes only a few minutes on modern computer hardware. In addition, this process can easily be parallelized, since the surrogate data sets can be processed independently, while collecting the $\langle z, c \rangle$ pairs from all surrogate data sets takes negligible time.

As an example of detecting assemblies with this procedure, we consider a trial with 100 parallel spike trains with an injected assembly of seven neurons (labeled {1, 2, 3, 4, 5, 6, 7}) and 7 coincidences, generated as described in section 7 for correlated data (20 Hz firing rate, duration 3 s, time bin length 3 ms, that is, 1000 time bins). We generate 10,000 surrogate data sets by means of spike-time randomization (see, e.g., Louis et al., 2010c; this simple approach is acceptable here due to the stationary process). The average number of closed patterns found with FIM in these surrogate data sets, subdivided by their size $z$ and number $c$ of coincidences (support) is shown in **Figure 10A**. After removing from the result on the original data all closed sets with signatures $\langle z, c \rangle$ that occur in surrogate data, we obtain the sets shown in **Table 3**.

This is a fairly typical result (although it was specifically selected to cover all relevant effects): the injected assembly is detected, but also some other sets, which, however, are all related to the injected assembly. Due to chance coincidences of neurons

---

[2]See, for example, fimi.ua.ac.be/src or www.borgelt.net/fpm.

outside of the assembly (here: neurons 35, 81, 85, 93) with some of the coincident spiking events of the assembly, we see a few supersets of the assembly, with lower support. Due to chance coincidences (resulting from background spikes) of some of the assembly neurons, we also see subsets of size 3 and 4, with a support exceeding the injected coincidences by 1 or 2. Finally, we see a set overlapping the assembly ({2, 3, 4, 35}), which results from neuron 35 firing together with four coincident spiking events of the assembly and the one additional coincident spiking event of the neurons {2, 3, 4} (which is caused by background spikes). Note that there are no sets that are unrelated to the injected assembly. How this result set can be reduced to the (most likely) assembly will be the topic of a subsequent paper (see Torre et al., in preparation).

However, compared to, say, Accretion, this result is already a huge improvement: running Accretion with a branching restriction of 2 on the same data yields (after removing duplicates— recall the redundancy of Accretion's search scheme) 66 sets, only 7 of which are related to the injected assembly (3 are supersets, 4 are overlapping patterns), while the assembly itself is not in the output. The unrelated patterns have sizes ranging from 2 to 5 neurons and exhibit between 2 and 9 coincidences. Without a branching restriction, Accretion yields even 105 sets, 30 of which are related to the injected assembly (5 supersets, 25 overlapping patterns).

In order to illustrate the detection capabilities of our method, **Figure 10** collects various results for data sets with injected assemblies with different sizes $z$ and coincidences $c$. **Figure 10B** shows



**FIGURE 10 | Assembly detection with surrogate data filtering based on size/coincidence signatures $\langle z, c \rangle$. (A)** decimal logarithm of the average number of patterns found in 10,000 surrogate data sets; **(B and C)** average numbers of patterns detected in 1000 data sets with injected assemblies with $\langle z, c \rangle = \langle 7, 7 \rangle$ and $\langle z, c \rangle = \langle 6, 6 \rangle$, respectively (gray squares indicate signatures that occur in the surrogates and are therefore eliminated); **(D)** false negatives for

injected assemblies with different $\langle z, c \rangle$ combinations; **(E–H)** decimal logarithm of the average number of patterns with different relations to the injected assembly: **(E)** proper supersets of the injected assembly, **(F)** proper subsets of the injected assembly, **(G)** patterns with at least two neurons from the injected assembly and at least one other neuron, **(H)** patterns having none or at most one neuron in common with the injected assembly.

**Table 3 | Closed frequent neuron sets found in data with an injected assembly with seven neurons and seven coincidences, after filtering with surrogate data; support/coincidences after the colon.**

| | | |
|---|---|---|
| $\{1, 2, 3, 4, 5, 6, 7, 81, 85\} : 2$ | $\{2, 3, 5, 6\} : 8$ | $\{2, 3, 7\} : 8$ |
| $\{1, 2, 3, 4, 5, 6, 7, 35\} : 4$ | $\{2, 3, 4, 35\} : 5$ | $\{2, 5, 7\} : 8$ |
| $\{1, 2, 3, 4, 5, 6, 7, 93\} : 3$ | $\{2, 3, 6\} : 9$ | $\{3, 4, 7\} : 8$ |
| $\{1, 2, 3, 4, 5, 6, 7\} : 7$ | $\{2, 3, 4\} : 8$ | |

the average number of patterns that are detected in data sets with an injected assembly with seven neurons and seven coincidences (averages over 1000 runs). **Figure 10C** shows analogous results for six neurons and six coincidences. **Figure 10D** shows the false negative rate (fraction of runs in which neither the assembly itself nor a superset is in the result set) for individual injected assemblies, covering all possible $\langle z, c \rangle$ combinations in $\{2, \ldots, 9\}^2$ (averages over 1000 runs). Note that false negatives occur basically only for $\langle z, c \rangle$ combinations that occur in surrogate data (see **Figure 10A**). Clearly, this is unavoidable.

The few false negatives represented by the blue squares are due to "clipping" caused by the time binning, which reduces the injected number of coincidences and thus creates a $\langle z, c \rangle$ pair that occurs in surrogate data. Note that for some signatures that occur in the surrogate data (e.g., $\langle z, c \rangle = \langle 8, 2 \rangle$ and $\langle z, c \rangle = \langle 3, 7 \rangle$) the injected assembly *is* sometimes detected, even though these signatures occur in surrogate data. The reason are chance coincidences, either of the whole set (for small $z$) or of an additional neuron with all coincident spiking events (for small $c$), which creates a pattern with a signature that is *not* eliminated by surrogate filtering.

**Figures 10E–H** give an idea of what and how many patterns are to be expected in the result for data with a single injected assembly, subdivided according to the size $z$ and the coincidences $c$ of the assembly. Not surprisingly, supersets become more frequent with an increasing number $c$ of coincident spiking events, subsets more frequent with an increasing size $z$ of the injected assembly. Overlapping patterns are much fewer (note the logarithmic scale!) and their number grows with both the size $z$ and the coincidences $c$. A clear benefit of our method is that it almost never produces patterns unrelated to the injected assembly (see **Figure 10H**): in the total of 64,000 data sets that the bar charts represent (1000 data sets per bar), only five patterns were detected that were *not* related to the assembly.

## 9. DISCUSSION

We started this paper by reviewing and analyzing the so-called Accretion method (Gerstein et al., 1978) for identifying neural assemblies from (discretized) parallel spike trains. Inspired by Accretion, we presented alternative assembly detection methods built on modifications and/or refinements of the two main constituents of Accretion: the statistical test to determine significance of the neuronal patterns built in the process and the subset conditions that a set of neurons needs to satisfy in order to be considered a neural assembly. Subset conditions alternative to those of Accretion were implemented with the help of FIM. By working on sets instead of sequences in a tree-like search space, FIM overcomes Accretion's redundancies and proves to be more efficient:

it yields shorter execution times even though the search is actually exhaustive, because no branching restriction is employed.

The results of our tests on trials with both independent spike trains (generated as independent Poisson processes) and correlated spike trains (generated as independent Poisson process with injected spike coincidences) showed high rates of false positives and false negatives for Accretion. Some of the FIM-based models that were built on alternative subset conditions and/or statistical tests performed better in terms of false positives but, generally, any such improvement was paid for by an increase in the number of false negatives. FIM alone, with no subset conditions and no statistical test, is not significantly worse than the other models since, although it yields the largest amount of false positives, it produces essentially no false negatives. Overall, the significance criteria employed by all of these models do not properly take into account the multiple testing problem, since they focus on tests of individual patterns. As a consequence, a high rate of false discoveries (false positives) and/or false negatives is always to be expected.

In the view of such results we proposed an alternative assembly detection method, based solely on FIM, which addresses the multiple testing problem properly. It is based on analyzing (with FIM) surrogate data that is generated from the original spike trains. The (closed) neurons sets that are found (with FIM) in the original data are then filtered by removing all patterns for which a counterpart occurs in surrogate data, that is, for which a pattern with the same size $z$ and the same number $c$ of coincidences (support) was found in some surrogate data set. The rationale underlying this approach is that a pattern with a counterpart in surrogate data could be a chance event and thus should not be considered significant.

Since we employ sophisticated and high-speed implementations of FIM algorithms to find the closed neuron sets, our method is efficient even though it requires to generate and analyze a substantial number of surrogate data sets. For example, generating and analyzing the 10,000 surrogate data sets underlying **Figure 10A** takes about 2 minutes on standard modern computer hardware, even without parallelization. This enables us to apply the method in a sliding window fashion in order to follow the dynamics of assembly activity.

This paper presented the basic approach of FIM and relevant statistics to detect and identify spike synchrony patterns in massively parallel spike data. In a subsequent paper (in preparation) we will report about further studies of dependencies on various analysis parameters (e.g., time bin size) and on features of the data [e.g., level of firing rates, various non-stationarities, deviations from Poisson processes etc. (Grün, 2009)]. However, our previous studies in other contexts (e.g., Louis et al., 2010b,c) and preliminary studies of our FIM-based method make us confident that we can account for such aspects by using surrogates that incorporate such features, e.g., local spike dithering or shift-surrogates (Gerstein, 2004; Pipa et al., 2008; Louis et al., 2010b).

Maximum-entropy models recently found application in the context of identification of the correlation structure in parallel spike trains. A general difference of methods based on maximum-entropy models in comparison to our approach is, that they evaluate if correlation structures exist at all by

comparing to non-existent correlations, whereas our approach aims to detect specific correlation structures. Although (Schneidman et al., 2006) claims that spike correlations found in experimental data are fully explained by pairwise correlations only Shlens et al. (2006), discusses in its study that existent higher-order correlations may be missed since they contribute only to a small percentage in the explanatory power. Roudi et al. (2009) showed in an extensive study that the estimates of the correlation structure estimated by maximum entropy models strongly depends on the existent parameters, in particular the size of the data set, and thus may lead to wrong conclusions on the correlation structure with a bias toward pairwise correlations. Thus we do not expect that maximum-entropy models would be able to extract assemblies of such small size of neurons and small number of occurrences of correlated spiking within a large number of neurons as it is possible with our approach.

In Feldt et al. (2009) a modification of Accretion was suggested as a way to improve its performance. The idea is to use the $p$-value of an independence test (e.g., Pearson's $\chi^2$) as the distance between two neurons (or sets of neurons) in a hierarchical agglomerative clustering algorithm. In general, such clustering starts with each of the given objects (here: neurons) in its own cluster. In each subsequent step it greedily merges the two clusters with the smallest distance, provided this distance is no greater than a user-specified maximum distance. For the Accretion variant, distances are measured by the $p$-value of an independence test and thus this maximum distance is simply the chosen significance level $\alpha$. Once no neuron groups can be merged any more, the method returns all clusters with more than one neuron. This method could be referred to as a "matrix version" of Accretion, because hierarchical agglomerative clustering is usually implemented as operating on a distance matrix, from which the rows (and corresponding columns) of the two clusters (here: sets of neurons) to be merged are removed and replaced by a row (and a column) representing the new cluster that resulted from the merger.

The matrix version of Accretion has the clear advantage that there is no branching. It executes at most $n - 1$ steps for $n$ neurons (after which all neurons would be merged into a single cluster) and cannot produce more than $\lfloor n/2 \rfloor$ clusters (unless singletons are counted as well). It may also be argued that it sometimes carries out more meaningful statistical tests, because it may merge two clusters both of which already contain two or more neurons, while Accretion always tests an already accreted group of neurons against a *single* new neuron.

However, these advantages are more than equalized by several disadvantages. Although the matrix version generally yields fewer results due to its greedy search scheme, it may still produce many false positives, because the problem of false positives lies mainly in the *potential* number of tests (i.e., the size of the search space) and the nature of the statistical tests and not in the *actual* number of tests that are executed. In addition, since the matrix variant also reports maximal sets, one almost never obtains an injected assembly exactly, but rather a superset (like in standard Accretion). Finally, as a hierarchical agglomerative clustering approach, it necessarily yields disjoint sets of neurons as (candidates of) assemblies. This leads to an unavoidable loss of results in case of overlapping assemblies, but may also rip an assembly apart if a bad merger is chosen due to the data characteristics, which is particularly likely when merging neurons into pairs.

The faster execution of the matrix version of Accretion is certainly attractive, but with the sophisticated, high-speed FIM implementations that are available, which make our method fairly efficient, there is no need to accept any of its drawbacks.

The only feature currently missing from our FIM-based approach is a way to reduce the found pattern set to the (most likely) assembly or assemblies. While a human can easily spot the actual assembly (or assemblies) by looking at the (usually reasonably small) output (see, for example, **Table 3**), an automatic method is desirable. We are currently working on a paper (Torre et al., in preparation) that presents and compares several suggestions of such pattern set reduction methods, which proved to be highly promising in preliminary experiments.

## ACKNOWLEDGMENTS

## SOFTWARE AND ADDITIONAL MATERIAL

The FIM library underlying the Python scripts with which we carried out our experiments is available at http://www.borgelt.net/pyfim.html

The actual Python and shell scripts as well as more extensive result diagrams are available at http://www.borgelt.net/accfim.html

## REFERENCES

Abdi, H. (2007). "Bonferroni and Šidák corrections for multiple comparisons," in *Encyclopedia of Measurement and Statistics*. ed N. J. Salkind (Thousand Oaks, CA: Sage Publications), 103–107.

Abeles, M. (1982). Role of the cortical neuron: integrator or coincidence detector? *Isr. J. Med. Sci.* 18, 83–92.

Abeles, M. (1991). *Corticonics: Neural Circuits of the Cerebral Cortex*. Cambridge: Cambridge University Press.

Abeles, M., and Gerstein, G. L. (1988). Detecting spatiotemporal firing patterns among simultaneously recorded single neurons. *J. Neurophysiol.* 60, 909–924.

Agrawal, R., Imielinski, T., and Swami, A. (1993). "Mining association rules between sets of items in large databasesm," *Proceedings 1993 ACM SIGMOD International Conference on Management of Data*, 207–216. New York, NY: ACM Press.

Benjamini, Y., and Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J. R. Stat. Soc. B Methodol.* 57, 289–300. doi: 10.2307/2346101

Berger, D., Warren, D., Normann, R., Arieli, A., and Grün, S. (2007). Spatially organized spike correlation in cat visual cortex. *Neurocomputing* 70, 2112–2116. doi: 10.1016/j.neucom.2006.10.141

Berger, D., Borgelt, C., Louis, S., Morrison, A., and Grün, S. (2010). Efficient identification of assembly neurons within massively parallel spike trains. *Comput.*

*Intell. Neurosci.* 2010:439648. doi:10.1155/2010/439648

Bonferroni, C. E. (1935). "Il calcolo delle assicurazioni su gruppi di teste," in *Studi in Onore del Professore Salvatore Ortu Carboni*, (Rome: Bardi), 13–60.

Borgelt, C. (2012). Frequent item set mining. *WIREs Data Mining Knowl. Discov.* 2, 437–456. doi: 10.1002/widm.1074

Buzsáki, G. (2004). Large-scale recording of neuronal ensembles. *Nat. Neurosci.* 7, 446–451. doi: 10.1038/nn1233

Dudoit, S., and van der Laan, M. J. (2008). *Multiple Testing Procedures with Application to Genomics*. New York, NY: Springer.

Feldt, S., Waddell, J., Hetrick, V. L., Berke, J. D., and Ochowski, M. (2009). Functional clustering algorithm for the analysis of dynamic network data. *Phys. Rev. E* 79:056104. doi: 10.1103/PhysRevE.79.056104

Fisher, R. A. (1925). *Statistical Methods for Research Workers*. Edinburgh: Oliver and Boyd.

Freiwald, W., Kreiter, A., and Singer, W. (1995). Stimulus dependent intercolumnar synchronization of single unit responses in Cat Area 17. *Neuroreport* 6, 2348–2352.

Fujisawa, S., Amarasingham, A., Harrison, M. T., and Buzsáki, G. (2008). Behavior-dependent short-term assembly dynamics in the medial prefrontal cortex. *Nat. Neurosci.* 11, 823–833. doi: 10.1038/nn.2134

Gansel, K. S., and Singer, W. (2012). Detecting multineuronal temporal patterns in parallel spike trains. *Front. Neuroinformatics* 6:18. doi: 10.3389/fninf.2012.00018

Gerstein, G. L., Perkel, D. H., and Subramanian, K. N. (1978). Identification of functionally related neural assemblies. *Brain Res.* 140, 43–62. doi: 10.1016/0006-8993(78)90237-8

Gerstein, G. L., and Aertsen, A. M. (1985). Representation of cooperative firing activity among simultaneously recorded neurons. *J. Neurophysiol.* 54, 1513–1528.

Gerstein, G. L. (2004). Searching for significance in spatio-temporal firing patterns. *Acta Neurobiol. Exp.* 2, 203–207.

Gerstein, G. L., Williamns, E. R., Diesmann, M., Grün, S., and Trengove, C. (2012). Detecting synfire chains in parallel spike data.

*J. Neurosci. Methods* 206, 54–64. doi: 10.1016/j.jneumeth.2012.02.003

Goethals, B. (2010). "Frequent set mining," in *Data Mining and Knowledge Discovery Handbook,* 2nd Edn. eds O. Maimon and L. Rokach (Berlin: Springer-Verlag), 321–338. doi: 10.1007/978-0-387-09823-4_16

Grün, S., Diesmann, M., and Aertsen, A. M. (2002). 'Unitary Events' in multiple single-neuron spiking activity. I. Detection and significance. *Neural Comput.* 14, 43–80. doi: 10.1162/089976602753284455

Grün, S. (2009). Data-driven significance estimation of precise spike correlation. *J. Neurophysiol.* 101, 1126–1140. doi: 10.1152/jn.00093.2008

Grün, S., and Rotter, S. (Eds.) (2010). *Analysis of Parallel Spike Trains*. Berlin: Springer-Verlag.

Hebb, D. O. (1949). *The Organization of Behavior*. New York, NY: J. Wiley and Sons.

Holm, S. (1979). A simple sequentially rejective multiple test procedure. *Scand. J. Stat.* 6, 65–70. doi: 10.2307/4615733

Kilavik, B. E., Roux, S., Ponce-Alvarez, A., Confais, J., Grün, S., and Riehle, A. (2009). Long-term modifications in motor cortical dynamics induced by intensive practice. *J. Neurosci.* 29, 12653–12663. doi: 10.1523/JNEUROSCI.1554-09.2009

Kohn, A., and Smith, M. A. (2005). Stimulus dependence of neuronal correlations in primary visual cortex of the macaque. *J. Neurosci.* 25, 3661–3673. doi: 10.1523/JNEUROSCI.5106-04.2005

König, P., Engel, A. K., and Singer, W. (1996). Integrator or coincidence detector? The role of the cortical neuron revisited. *Trends Neurosci.* 19, 130–137. doi: 10.1016/S0166-2236(96)80019-1

Kuhn, A., Aertsen, A., and Rotter, S. (2003). Higher-order statistics of input ensembles and the response of simple model neurons. *Neural Comput.* 15, 67–101. doi: 10.1162/089976603321043702

Louis, S., Borgelt, C., and Grün, S. (2010a). Complexity distribution as a measure for assembly size and temporal precision. *Neural Netw.* 23, 705–712. doi: 10.1016/j.neunet.2010.05.004

Louis, S., Gerstein, G. L., Grün, S., and Diesmann, M. (2010b). Surrogate spike train generation through dithering in operational time. *Front. Comput. Neurosci.* 4:127. doi: 10.3389/fncom.2010.00127

Louis, S., Borgelt, C., and Grün, S. (2010c). "Generation and selection of surrogate methods for correlation analysis," in *Analysis of Parallel Spike Trains*, eds S. Grün and S. Rotter (Berlin: Springer-Verlag), 359–382. doi: 10.1007/978-1-4419-5675-0_17

Masud, M., and Borisyuk, R. (2011). Statistical technique for analysing functional connectivity of multiple spike trains. *J. Neurosci. Methods* 196, 201–219. doi: 10.1016/j.jneumeth.2011.01.003

Nawrot, M., Aertsen, A., and Rotter, S. (1999). Single-trial estimation of neuronal firing rates: from single neuron spike trains to population activity. *J. Neurosci. Methods* 94, 81–92. doi: 10.1016/S0165-0270(99)00127-2

Nicolelis, M. A., Ghazanfar, A. A., Faggin, B. M., Votaw, S., and Oliveira, L. M. (1997). Reconstructing the engram: simultaneous, multisite, many single neuron recordings. *Neuron* 18, 529–537. doi: 10.1016/S0896-6273(00)80295-0

Pipa, G., Wheeler, D. W., Singer, W., and Nikolic, D. (2008). Neuroxidence: reliable and efficient analysis of an excess or deficiency of joint-spike events. *J. Neurosci. Methods* 25, 64–88. doi: 10.1007/s10827-007-0065-3

Riehle, A., Grün, S., Diesmann, M., and Aertsen, A. M. (1997). Spike synchronization and rate modulation differentially involved in motor cortical function. *Science* 278, 1950–1953. doi: 10.1126/science.278.5345.1950

Roudi, Y., Nirenberg, S., and Latham, P. E. (2009). Pairwise maximum entropy models for studying large biological systems: when they can work and when they can't. *PLoS Comput. Biol.* 5:e1000380. doi: 10.1371/journal.pcbi.1000380

Schneidman, E., Berry, M. J., Segev, R., and Bialek, W. (2006). Weak pairwise correlations imply strongly correlated network states in a neural population. *Nature* 440, 1007–1012. doi: 10.1038/nature04701

Schrader, S., Grün, S., Diesmann, M., and Gerstein, G. L. (2008). Detecting synfire chain activity using massively parallel spike train recording. *J. Neurophysiol.* 100, 2165–2176. doi: 10.1152/jn.01245.2007

Schultze-Kraft, M., Diesmann, M., Grün, S., and Helias, M. (2013). Noise suppression and surplus synchrony by coincidence detection.

*PLoS Comput. Biol.* 9:e1002904. doi: 10.1371/journal.pcbi.1002904

Shimazaki, H., Amari, S.-I., Brown, E. N. B., and Grün, S. (2012). State-space analysis of time-varying higher-order spike correlation for multiple neural spike train data. *PLoS Comput. Biol.* 8:e1002385 doi: 10.1371/journal.pcbi.1002385

Shlens, J., Field, G. D., Gauthier, J. L., Grivich, M. I., Petrusca, D., Sher, A, et al. (2006). The structure of multi-neuron firing patterns in primate retina. *J. Neurosci.* 26, 8254–8266. doi: 10.1523/JNEUROSCI.1282-06.2006

Staude, B., Grün, S., and Rotter, S. (2010a). Higher-order correlations in non-stationary parallel spike trains: statistical modeling and inference. *Front. Comput. Neurosci.* 4:16. doi: 10.3389/fncom.2010.00016

Staude, B., Rotter, S., and Grün, S. (2010b). CuBIC: cumulant based inference of higher-order correlations in massively parallel spike trains. *J. Comput. Neurosci.* 29, 327–350. doi: 10.1007/s10827-009-0195-x

Vaadia, E., Haalman, I., Abeles, M., Bergman, H., Prut, Y., Slovin, H., et al. (1995). Dynamics of neuronal interactions in monkey cortex in relation to behavioural events. *Nature* 373, 515–518. doi: 10.1038/373515a0