

Quantifying spike train synchrony and directionality: Measures and Applications

Thomas Kreuz^{a,b}

^a*Institute for Complex Systems (ISC), National Research Council (CNR), Sesto Fiorentino, Italy*

^b*National Institute of Nuclear Physics (INFN), Florence Section, Sesto Fiorentino, Italy*

Abstract

By introducing the twin concepts of reliability and precision along with the corresponding measures, Mainen and Sejnowski's seminal 1995 paper "Reliability of spike timing in neocortical neurons" [1] paved the way for a new kind of quantitative spike train analysis. In subsequent years a host of new methods was introduced that measured both the synchrony among neuronal spike trains and the directional component, e.g. how activity propagates between neurons. This development culminated with a new class of measures that are both time scale independent and time resolved. These include the two spike train distances ISI- and SPIKE-Distance as well as the coincidence detector SPIKE-Synchronization and its directional companion SPIKE-Order. This article will not only review all of these measures but also include two recently proposed algorithms for latency correction which build on SPIKE-order and aim to optimize the spike time alignment of sparse spike trains with well-defined global spiking events. For the sake of clarity, all these methods will be illustrated on artificially generated data but in each case exemplary applications to real neuronal data will be described as well.

1. Introduction

Mainen and Sejnowski's 30 year old paper [1] celebrated here in this special issue has not only inspired a multitude of experimental and theoretical investigations on a wide range of topics such as the mechanisms of spike generation and the intricacies of neural coding, it also started the development of a plethora of new methods to quantify spike train synchrony.

Measuring the degree of synchrony within a set of spike trains is a common task in two major scenarios. Spike trains are recorded either simultaneously from a population of neurons, or in successive time windows from only one neuron. In this latter scenario, repeated presentation of the same stimulus addresses the reliability of individual neurons (as in [1]), while different stimuli are used to investigate neural coding and to find the features of the response that provide the optimal discrimination [2, 3]. These two applications are related since for a good clustering performance one needs not only a pronounced discrimination between different stimuli (low inter-stimulus spike train synchrony) but also a large reliability for the same stimulus (high intra-stimulus spike train synchrony).

In [1] itself, the demonstration of reliable and precise spike initiation in the neocortex was achieved by means of two measures based on the post-stimulus time histogram (PSTH), which, fittingly, were termed reliability and precision. Just a year after the spike train metric by Victor and Purpura [4] was proposed which evaluates the cost needed to transform one spike train into the other, using only certain elementary steps. This was soon followed by the van Rossum metric [5] which measures the Euclidean distance between two spike trains after convolution of the spikes with an exponential function. Other

approaches from that time quantify the cross correlation of spike trains after exponential [6] or Gaussian filtering [7], or exploit the exponentially weighted distance to the nearest neighbor in the other spike train [8].

A commonality to all of these measures is the existence of one parameter that sets the time scale for the analysis. By contrast, in 2007 the parameter-free ISI-Distance was introduced and compared with these existing approaches [9, the very first citation of which happens to be [1]]. This was the first of a new class of measures that is not only time scale independent but also time-resolved. In fact, it is this class of measures (which also includes the SPIKE-Distance, SPIKE-Synchronization, SPIKE-Order, and Spike Train Order) as well as algorithms and applications derived from them that is the topic to be reviewed in this contribution to the special issue.

First there are the ISI-Distance [9] and the SPIKE-Distance [10], two spike train distances (i.e., measures inverse to synchrony) that focus on instantaneous comparisons of firing rate and spike timing, respectively (Chapter 2). A complementary family of measures is given by SPIKE-Synchronization [11], a sophisticated coincidence detector that quantifies the level of synchrony from the number of quasi-simultaneous appearances of spikes, and its directional variants, SPIKE-Order and Spike Train Order [12] which allow to sort multiple spike trains from leader to follower and to quantify the consistency of the temporal leader-follower relationships for both the original and the optimized sorting (Chapter 3). Building on this SPIKE-Synchronization and Spike Train Order framework, in Chapter 4 algorithms are presented which perform latency correction, i.e., optimize the spike time alignment of sparse spike trains with well-defined global spiking events, both for events without [13] and with overlap [14]. Finally, a short outlook is given in Chapter 5.

Email address: thomas.kreuz@cnr.it (Thomas Kreuz)

Throughout this article the number of spike trains is denoted by N , indices of spike trains by n and m , spike indices by i and j and the number of spikes in spike train n by M_n . The spike times of spike train n are thus given as $\{t_i^{(n)}\}$ with $i = 1 \dots M_n$. Without loss of generality the interval under consideration is defined to last from time $t = 0$ to $t = T$.

2. ISI-Distance and SPIKE-Distance

The first step in calculating both the ISI-Distance D_I [9] and the SPIKE-Distance D_S [10] is to transform the sequences of discrete spike times into (quasi-)continuous temporal dissimilarity profiles with one value for each time instant. The temporal profile $I(t)$ of the ISI-Distance is derived from the interspike intervals, while for the SPIKE-Distance the profile $S(t)$ is calculated from differences between the spike times of the spike trains.

Useful for both definitions and starting with just two spike trains, to each spike train $n = 1, 2$ and each time instant t (Fig. 1A) are assigned three piecewise constant quantities, the time of the previous spike

$$t_P^{(n)}(t) = \max(t_i^{(n)} | t_i^{(n)} \leq t), \quad (1)$$

the time of the following spike

$$t_F^{(n)}(t) = \min(t_i^{(n)} | t_i^{(n)} > t), \quad (2)$$

and the interspike interval

$$x_{\text{ISI}}^{(n)}(t) = t_F^{(n)}(t) - t_P^{(n)}(t). \quad (3)$$

The ambiguity regarding the definition of the very first and the very last interspike interval and the special cases of empty spike trains or spike trains with just one spike are dealt with in [15].

2.1. ISI dissimilarity profile

The ISI-Distance [9] and its multivariate extension [16] are based on the instantaneous interspike intervals (see Fig. 1A). A time-resolved, symmetric, and time scale adaptive measure of the relative firing rate pattern is obtained by calculating the normalized instantaneous ratio between $x_{\text{ISI}}^{(n)}$ and $x_{\text{ISI}}^{(m)}$ as

$$I(t) = \frac{|x_{\text{ISI}}^{(n)}(t) - x_{\text{ISI}}^{(m)}(t)|}{\max\{x_{\text{ISI}}^{(n)}(t), x_{\text{ISI}}^{(m)}(t)\}}. \quad (4)$$

This ISI dissimilarity profile becomes 0 for identical ISI in the two spike trains, and approaches 1 whenever one spike train has a much higher firing rate than the other. As the interspike intervals are piecewise constant functions, also the dissimilarity profile is piecewise constant. The ISI dissimilarity profile for an artificial example of 50 spike trains with global events of slowly varying levels of jitter and different noise levels (see Fig. 2A) is shown in Fig. 2B.

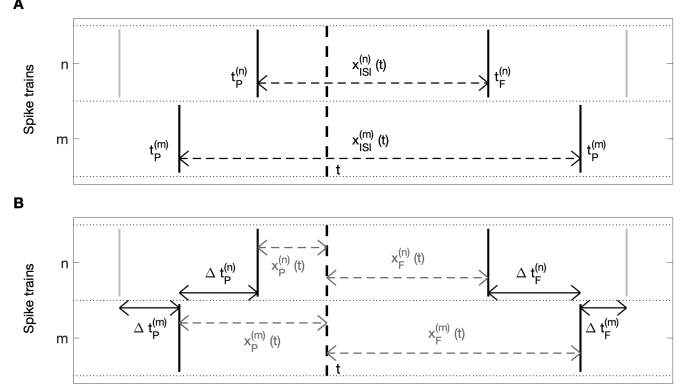


Figure 1: Schematic illustration of how the ISI-Distance D_I and the SPIKE-Distance D_S are derived from local quantities around an arbitrary time instant t . A. The ISI dissimilarity profile $I(t)$ is calculated from the instantaneous interspike intervals. B. Additional spike-based variables make the SPIKE dissimilarity profile $S(t)$ sensitive to spike timing. Modified from [11].

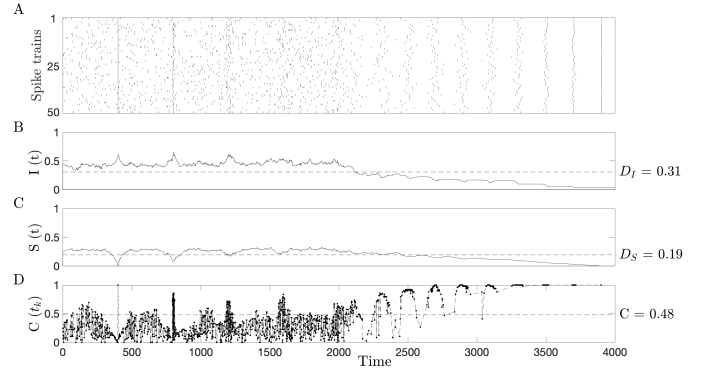


Figure 2: Artificial example dataset of 50 spike trains (A) and the corresponding profiles for the ISI-Distance (B), the SPIKE-Distance (C), and SPIKE-Synchronization (D), as well as their average values. In the first half within the noisy background there are 4 regularly spaced global events with increasing jitter. The second half consists of 10 global events with decreasing jitter but without any noisy background. Modified from [17].

2.2. SPIKE dissimilarity profile

Since the ISI dissimilarity profile is based on the relative length of simultaneous interspike intervals, it is optimally suited to quantify similarities in firing rate profiles. However, it is not designed to track the type of synchrony that is mediated by spike timing (see also [18]). This is a particular kind of sensitivity which is not only of theoretical but also of high practical importance since coincidences of spikes occur in many different neuronal circuits such as the visual cortex [19, 20] or the retina [21]. This led to the development of the SPIKE dissimilarity profile which uniquely combines some of the useful properties of the ISI dissimilarity profile with a specific focus on spike timing. The definition presented here is the one introduced in [10], which improves considerably on the original proposal [22].

To derive the instantaneous dissimilarity profile for the SPIKE-Distance, in the beginning to each time instant four corner spikes are assigned (see Fig. 1B): the preceding spike of spike train n , $t_P^{(n)}$, the following spike of spike train n , $t_F^{(n)}$, the preceding spike of spike train m , $t_P^{(m)}$, and, finally, the following

spike of spike train m , $t_F^{(m)}$. Each of these four corner spikes can then be attached with the spike time difference to the nearest spike in the other spike train, e.g., for the previous spike of spike train n ,

$$\Delta t_P^{(n)}(t) = \min(|t_P^{(n)}(t) - t_i^{(m)}|) \quad (5)$$

and analogously for $t_F^{(n)}$, $t_P^{(m)}$, and $t_F^{(m)}$.

Subsequently, for each spike train separately a locally weighted average is employed such that the difference of the closer spike dominates: The weighting factors are the intervals from the time instant under consideration to its previous and to its following spike, e.g.,

$$x_P^{(n)}(t) = t - t_P^{(n)}(t) \quad (6)$$

and

$$x_F^{(n)}(t) = t_F^{(n)}(t) - t. \quad (7)$$

Accordingly, for that spike train the local weighting of the two spike time differences reads:

$$S_n(t) = \frac{\Delta t_P^{(n)}(t)x_F^{(n)}(t) + \Delta t_F^{(n)}(t)x_P^{(n)}(t)}{x_{ISI}^{(n)}(t)}. \quad (8)$$

Averaging over the contributions of both spike trains and normalizing by the mean interspike interval gives the dissimilarity profile of the rate-independent SPIKE-Distance (which was proposed in [15]):

$$S_{RI}(t) = \frac{S_n(t) + S_m(t)}{x_{ISI}^{(n)}(t) + x_{ISI}^{(m)}(t)}. \quad (9)$$

This quantity takes into account relative distances within each spike train, but ignores differences in time scale between spike trains. In order to account for differences in firing rate and get these ratios straight, in a last step the contributions from the two spike trains are locally weighted by their instantaneous interspike intervals. This defines the SPIKE dissimilarity profile:

$$S(t) = \frac{S_n(t)x_{ISI}^{(m)}(t) + S_m(t)x_{ISI}^{(n)}(t)}{\frac{1}{2} [x_{ISI}^{(n)}(t) + x_{ISI}^{(m)}(t)]^2}. \quad (10)$$

Since this dissimilarity profile is obtained from a linear interpolation of piecewise constant quantities, it is itself piecewise linear (with potential discontinuities at the spikes). The SPIKE dissimilarity profile of the same artificial spike train set used in Section 2.1 is shown in Fig. 2C.

2.3. Similarities and differences

Both the ISI- and the SPIKE-Distance are defined as the temporal average of the respective time profile, e.g., for the ISI-Distance,

$$D_I = \frac{1}{T} \int_0^T dt I(t). \quad (11)$$

Also, for both distances, there exist two ways to derive from the bivariate versions of the profiles the multivariate extension

to $N > 2$ spike trains. First it can be obtained by simply averaging over only the upper right triangular part (since both distances are symmetric) of the pairwise distance matrix, here again for the ISI-Distance,

$$D_I = \frac{2}{N(N-1)} \sum_{n=1}^{N-1} \sum_{m=n+1}^N D_I^{n,m}. \quad (12)$$

In this case the temporal average of Eq. 11 is followed by the (spatial) average over all pairs of spike trains of Eq. 12. However, these two averages commute, so it is also possible to achieve the same kind of time-resolved visualization as in the bivariate case by first calculating the instantaneous average $S^a(t)$ (now for the SPIKE-Distance) over all pairwise instantaneous values $S^{n,m}(t)$:

$$S^a(t) = \frac{2}{N(N-1)} \sum_{n=1}^{N-1} \sum_{m=n+1}^N S^{n,m}(t). \quad (13)$$

This time the spatial average is followed by the temporal average

$$D_S = \frac{1}{T} \int_0^T dt S^a(t), \quad (14)$$

and so Eqs. 12 and 14 yield the exact same value.

Both dissimilarity profiles, the piecewise constant $I(t)$ and the piecewise linear $S(t)$, as well as both distances D_I and D_S (calculated according to Eq. 11) always stay within the interval $[0, 1]$. For the SPIKE-Distance the limit value $D_S = 0$ is obtained only for perfectly identical spike trains, while for the ISI-distance $D_I = 0$ can also be attained for periodic spike trains with exactly the same period. Further mathematical properties of both distances (including expectation values for Poisson spike trains) are derived in [23].

Spike trains can be analyzed on many different spatial and temporal scales, accordingly these two time-resolved and pairwise measures of spike train dissimilarity allow for several levels of information extraction [11]. In the most detailed representation one instantaneous value is obtained for each pair of spike trains, while the most condensed representation of successive temporal and spatial averaging leads to one single distance value that describes the overall level of synchrony for a spike train set over a given time interval. In between these two extremes are spatial averages (multivariate dissimilarity profiles, see Fig. 2, B and C) and temporal averages (pairwise dissimilarity matrices, examples for the SPIKE-Distance are shown in Fig. 3). A movie version of these matrices can be found in the Supplemental Material of [10] and in the playlist of the SPIKY-channel on Youtube.

ISI-Distance - Example applications

Since its proposal in 2007, the ISI-Distance has been applied to electrophysiological data close to 100 times (For a constantly updated lists of such studies for all the measures dealt with here please refer to <https://www.thomaskreuz.org/publications/isi-spike-articles>). Here are a few examples: Recently,

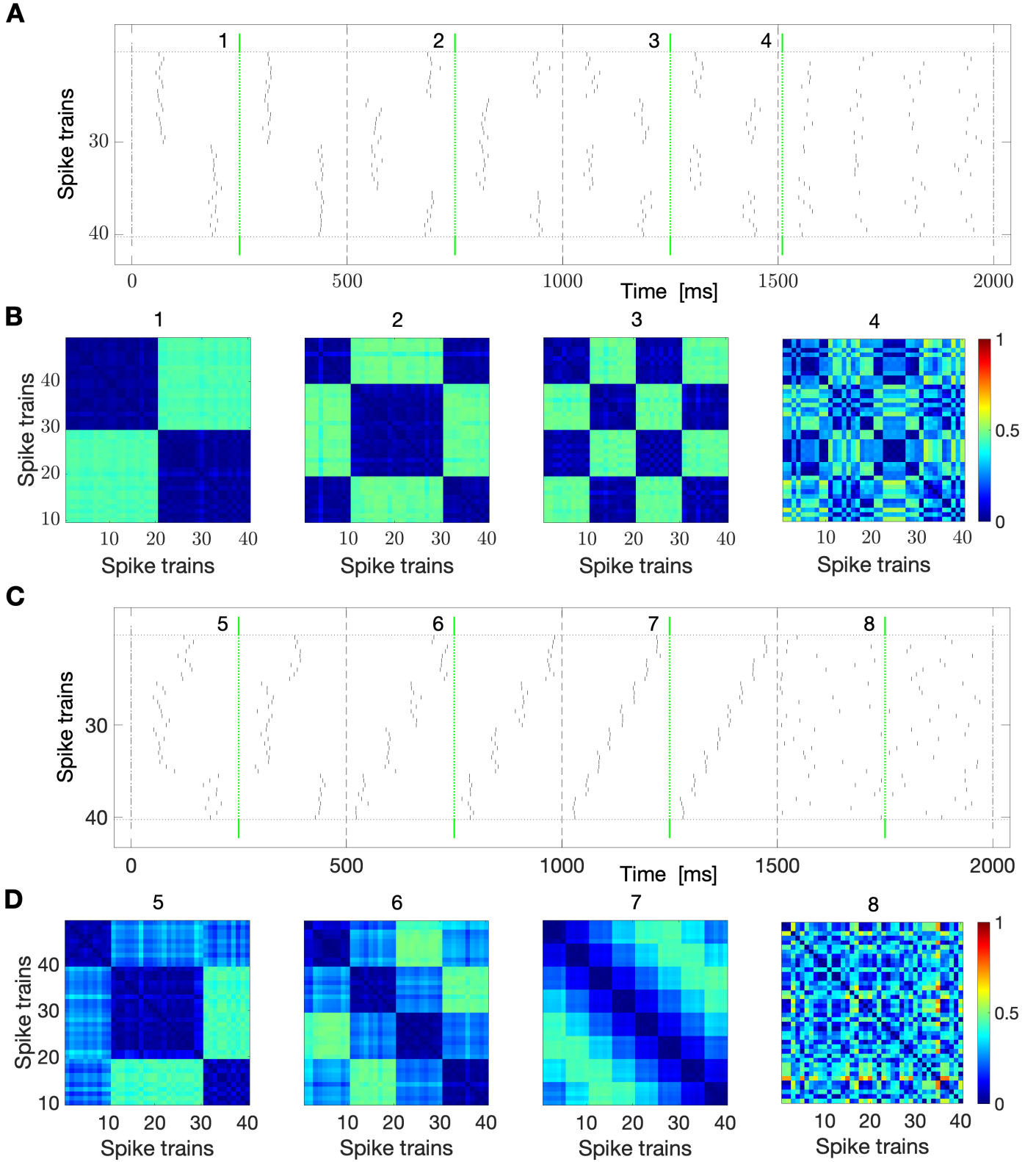


Figure 3: Instantaneous clustering for artificially generated spike trains (A, C) whose clustering behavior changes every 500 ms: from four different variants of two clusters via three, four and eight clusters to random spiking. B, D. Matrices of pairwise instantaneous SPIKE-dissimilarity values for the time instants marked by the green lines in A and C, respectively. Modified from [10].

the ISI-Distance was used as an important feature for simulation-based inference (SBI), a machine learn-

ing approach that automatically estimates parameters that replicate the activity of "human induced pluripo-

tent stem cells" (hiPSCs)-derived neuronal networks on multi-electrode arrays (MEAs) [24]. Other studies classified Alzheimer's disease phenotype based on hippocampal electrophysiology [25], assessed the performance of biologically-inspired image processing in computational retina models [26], and performed spike train synchrony analysis of neuronal cultures [27].

SPIKE-Distance - Example applications

Even though the SPIKE-Distance was proposed six years after the ISI-Distance, it has already surpassed its counterpart in usage (which again seems to indicate that spike timing matters). The most recent of the more than 100 applications shows that motor cortex stimulation increases the variability of single-unit spike configuration in parkinsonian but not in normal rats [28]. Other studies evaluate the coordinated activity in human induced pluripotent stem cells (iPSCs) derived neuron-astrocyte co-cultures [29], or perform a non-parametric physiological classification of retinal ganglion cells in the mouse retina [30]. Interestingly, the SPIKE-Distance, like all the other measures, is often used in many different fields and contexts outside of neuroscience, for example to assess the reproducibility of eyeblink timing during formula car driving [31].

On the other hand, the rate-independent SPIKE-Distance has recently been employed to quantify neural discrimination of different frequency tones in a rat model of Fragile X Syndrome, a leading inherited cause of autism spectrum disorders (ASD) [32]. It has also helped to dissect the contributions of parvalbumin neurons towards rate and timing-based coding in complex auditory scenes and to explore their ability to reduce cortical noise [33].

3. SPIKE-Synchronization and SPIKE-Order

3.1. Spike matching via adaptive coincidence detection

As Fig. 4A illustrates, in general it is basically impossible to judge whether two spikes are coincident or not without taking the local context into account. To overcome this problem, (author?) [34] proposed an adaptive coincidence detection which is scale- and thus parameter-free since the minimum time lag $\tau_{ij}^{(n,m)}$ up to which two spikes $t_i^{(n)}$ and $t_j^{(m)}$ from different spike trains are considered to be synchronous is adapted to the local firing rates (higher

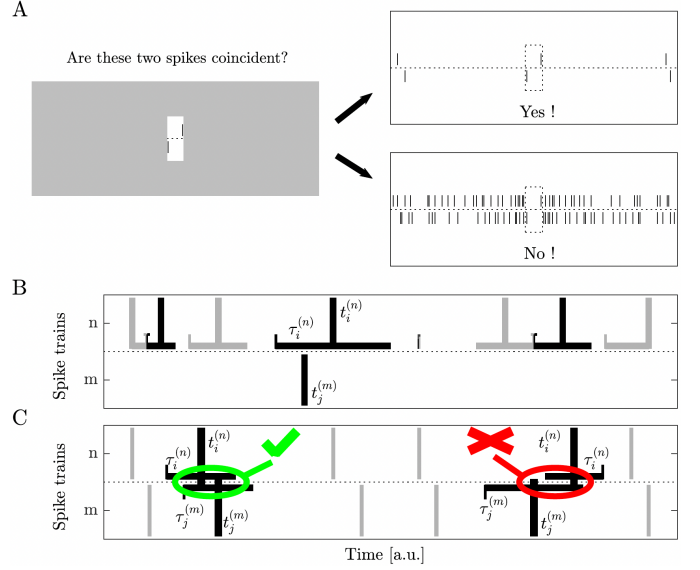


Figure 4: A. Motivation for adaptive coincidence detection. Depending on local context the same two spikes (left) can appear as coincident (right, top) or as non-coincident (right, bottom). B/C. Illustration of adaptive coincidence detection. The first step (B) assigns to each spike $t_i^{(n)}$ of spike train n a potential coincidence window that does not overlap with any other coincidence window: $\tau_i^{(n)} = \min\{t_{i+1}^{(n)} - t_i^{(n)}, t_i^{(n)} - t_{i-1}^{(n)}\}/2$. Thus any spike from spike train m can at most be coincident with one spike from spike train n . Short vertical lines mark the times right in the middle between two spikes. For better visibility spikes and their coincidence windows are shown in alternating bright and dark color. In the same way (C) a coincidence window $\tau_j^{(m)} = \min\{t_{j+1}^{(m)} - t_j^{(m)}, t_j^{(m)} - t_{j-1}^{(m)}\}/2$ is defined for spike $t_j^{(m)}$ from spike train m . For two spikes to be coincident they have to be in each other's coincidence window which means that their absolute time difference has to be smaller than $\tau_{ij} = \min\{\tau_i^{(n)}, \tau_j^{(m)}\}$ (which is equivalent to Eq. 15). In this example the two spikes on the left are coincident, whereas the two spikes on the right are not. Modified from [12].

firing rates lead to smaller coincidence windows):

$$\tau_{ij}^{(n,m)} = \min\{t_{i+1}^{(n)} - t_i^{(n)}, t_i^{(n)} - t_{i-1}^{(n)}, t_{j+1}^{(m)} - t_j^{(m)}, t_j^{(m)} - t_{j-1}^{(m)}\}/2. \quad (15)$$

Starting with a pair of spikes from two different spike trains (the minimal case for which there can be a coincidence or not) the adaptive coincidence criterion can then be applied in a multivariate context [11] by defining for each spike i of spike train n a coincidence indicator (which considers all spikes j of spike train m):

$$C_i^{(n,m)} = \begin{cases} 1 & \text{if } \min_j (|t_i^{(n)} - t_j^{(m)}|) < \tau_{ij}^{(n,m)} \\ 0 & \text{otherwise.} \end{cases} \quad (16)$$

Due to the minimum function and the "<" (instead of " $\leq$ ") any spike can at most be coincident with one spike (the nearest one) in the other spike train (Fig. 4B) and thereby an unambiguous spike matching is

guaranteed. The coincidence indicator $C_i^{(n,m)}$ is either 1 or 0 depending on whether the spike i of spike train n is part of a coincidence with any spike of spike train m or not (Fig. 4C).

3.2. SPIKE-Synchronization

In order to derive an overall measure of spike matching, this adaptive criterion of "closeness in time" is applied to all spikes of a given spike train set. By averaging over all $N-1$ bivariate coincidence indicators involving spike i of spike train n , a multivariate normalized coincidence counter is obtained:

$$C_i^{(n)} = \frac{1}{N-1} \sum_{m \neq n} C_i^{(n,m)}. \quad (17)$$

Subsequently, pooling the coincidence counters of the whole spike train set results in a single multivariate SPIKE-Synchronization profile

$$\{C(t_k)\} = \bigcup_n \{C_{i(k)}^{(n(k))}\}, \quad (18)$$

where the spike indices $i(k)$ and the spike train indices $n(k)$ are mapped onto a global spike index k .

With $M = \sum_{n=1}^N M_n$ denoting the total number of spikes, the average of this profile yields the SPIKE-Synchronization

$$C = \begin{cases} \frac{1}{M} \sum_{k=1}^M C(t_k) & \text{if } M > 0 \\ 1 & \text{otherwise,} \end{cases} \quad (19)$$

the overall fraction of coincidences in the whole spike train set [11]. SPIKE-Synchronization attains the value 0 if and only if there are no coincidences at all and reaches the value 1 if and only if each spike in every spike train has one matching spike in all the other spike trains - or if there are no spikes at all (since common silence can also be considered as perfect synchrony). The profile for the artificial example from Section 2.1 is shown in Fig. 2D, and a discussion of the mathematical properties of SPIKE-Synchronization can again be found in [23].

SPIKE-Synchronization - Example applications

SPIKE-Synchronization was proposed only in 2015, nevertheless it has already been employed more than 50 times. In two very recent publications SPIKE-Synchronization has been used to analyze short-term plasticity in excitatory-inhibitory networks [35] and to identify the optimal point for

transitioning from one gait to another in legged robot locomotion [36]. In other studies SPIKE-Synchronization has been applied as machine learning feature in order to discriminate the three states of a network of stochastic spiking neurons [37] or to show that birds multiplex spectral and temporal visual information via retinal on- and off-channels [38].

3.3. SPIKE-Order

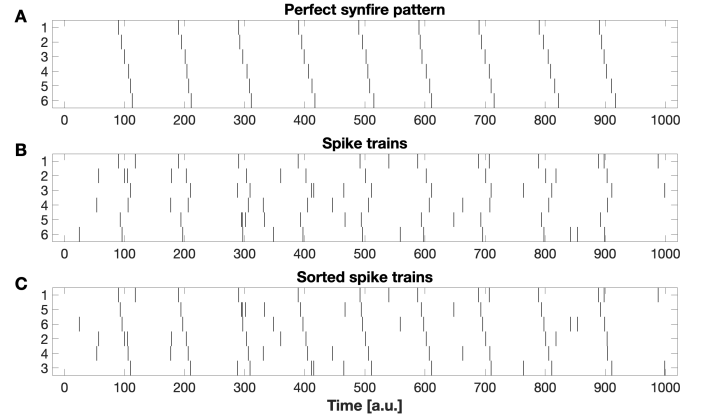


Figure 5: Using the Spike Train Order framework to sort spike trains from leader to follower. A. Perfect Synfire pattern. B. Unsorted set of spike trains. C. The same spike trains as in B but now sorted. Modified from [12].

Often a set of spike trains repeatedly exhibits well-defined patterns of spatio-temporal propagation where activity first appears at a specific location and then spreads to other areas until potentially becoming a global event. If a set of spike trains exhibits perfectly consistent repetitions of the same global propagation pattern, this is called a *synfire pattern* (for an example see Fig. 5A). On the other hand, for any given spike train set containing propagation patterns the question arises whether there is spatiotemporal consistency in these patterns, i.e., to what extent do they resemble a synfire pattern, and are there spike trains that consistently lead global events and others that invariably follow these leaders?

The symmetric measure SPIKE-Synchronization (Section 3.2) is invariant to which of the two spikes in a coincidence pair is the leader and which is the follower. To take into account the temporal order of the spikes, the directional measures SPIKE-Order and Spike Train Order [12] are needed. Building on SPIKE-Synchronization, the Spike Train Order framework allows to sort the spike trains from leader to follower (compare Figs. 5B and 5C) and to evalu-

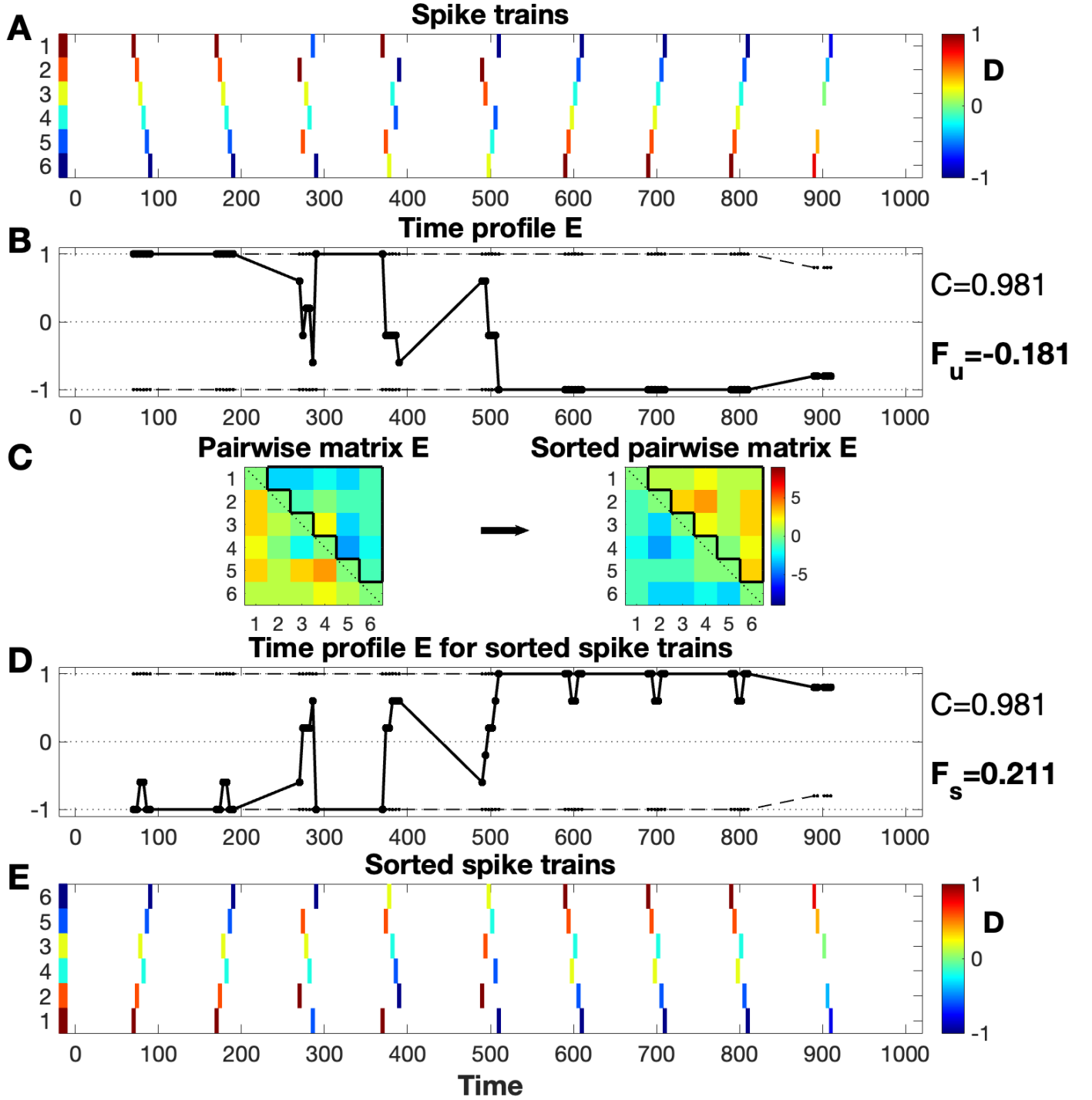


Figure 6: Spike Train Order for an artificial dataset consisting of 6 spike trains arranged into nine reliable events. The first two events are in order, the last four in inverted order, and for the three events in between the order is random. In the last event one spike is missing. A. Unsorted spike trains with the spikes color-coded according to the value of the SPIKE-Order $D(t_k)$. B. Spike Train Order profile $E(t_k)$. Events with different firing order can clearly be distinguished. The Synfire Indicator F_u for the unsorted spike trains is slightly negative reflecting the predominance of the inversely ordered events. C. Pairwise cumulative Spike Train Order matrix E before (left) and after (right) sorting. The optimal order maximizes the upper right triangular matrix $E_{<}$ (Eq. 26), marked in black. The arrow in between the two matrices indicates the sorting process. D. Spike Train Order profile $E(t_k)$ and its average value, the Synfire Indicator F_s for the sorted spike trains (shown in subplot E). Modified from [12].

ate the consistency of the preferred order via a measure called the Synfire Indicator. The application of the whole procedure to a rather simple example

dataset can be traced in Fig. 6.

First, the spike in spike train m that matches spike

i in spike train n is identified as

$$j' = \arg \min_j (|t_i^{(n)} - t_j^{(m)}|). \quad (20)$$

Subsequently, the bivariate anti-symmetric SPIKE-Order

$$\begin{aligned} D_i^{(n,m)} &= C_i^{(n,m)} \cdot \text{sign}(t_{j'}^{(m)} - t_i^{(n)}) \\ D_{j'}^{(m,n)} &= C_{j'}^{(m,n)} \cdot \text{sign}(t_i^{(n)} - t_{j'}^{(m)}) = -D_i^{(n,m)}, \end{aligned} \quad (21)$$

assigns to each spike either a +1 or a -1 depending on whether it is leading or following the coincident spike in the other spike train (and accordingly for that coincident spike). If the two spikes occur at exactly the same time, they both get a zero.

Since SPIKE-Order distinguishes leading and following spikes, it is used to color-code the individual spikes on a leader-to-follower scale (see, e.g., Fig. 6A). However, its profile is invariant under exchange of spike trains and thus it looks the same for all events, independent of the order of the spikes within the event. Moreover, averaging over the D -profile values of all spikes (which is equivalent to averaging over all coincidences) necessarily leads to a mean value of 0, since in each coincidence for every leading spike (+1) there has to be a following spike (-1).

Spike Train Order E is similar to SPIKE-Order D but there are two important differences: First, this value depends on the order of the spike trains (and not on the order of the spikes), and second, it is symmetric, so both spikes are assigned the same value:

$$E_i^{(n,m)} = C_i^{(n,m)} \cdot \begin{cases} \text{sign}(t_{j'}^{(m)} - t_i^{(n)}) & \text{if } n < m \\ \text{sign}(t_i^{(n)} - t_{j'}^{(m)}) & \text{if } n > m \end{cases} \quad (22)$$

and

$$E_{j'}^{(m,n)} = E_i^{(n,m)}. \quad (23)$$

In particular, Spike Train Order assigns to both spikes a +1 (-1) in case the two spikes are in the correct (wrong) order, i.e., the spike from the spike train with the lower spike train index is the leader (the follower). Once more the value 0 is obtained when the time of the two coincident spikes is absolutely identical (but also when there is no coincident spike in the other spike train).

The multivariate profile $E(t_k)$, again derived according to Eq. 18, is also normalized between 1 and -1 and the extreme values belong to a completely coincident event with all spikes emitted in the correct

(incorrect) order from first (last) to last (first) spike train, respectively (compare the first two versus the last four events in Fig. 6B). The value 0 is obtained either if a spike is not part of any coincidence or if the order is such that correctly and incorrectly ordered spike train pairs cancel each other.

By construction (Eqs. 21 and 22), C_k is an upper bound for the absolute value of both D_k and E_k . In contrast to the SPIKE-Order profile D_k , for the Spike Train Order profile E_k it does make sense to calculate the average value. In fact, this is the first way to define the Synfire Indicator (which is described in more detail just below):

$$F = \frac{1}{M} \sum_{k=1}^M E(t_k). \quad (24)$$

The important task of sorting the spike trains from leader to follower can be achieved via the cumulative and anti-symmetric Spike Train Order matrix

$$E^{(n,m)} = \sum_i E_i^{(n,m)} \quad (25)$$

which quantifies the temporal relationship between the spikes in spike trains n and m . If $E^{(n,m)} > 0$, spike train n is leading spike train m (on average), while $E^{(n,m)} < 0$ implies m is the leading spike train. For a Spike Train Order in line with the synfire property (i.e., exhibiting consistent repetitions of the same global propagation pattern), $E^{(n,m)} > 0$ for all $n < m$ (and accordingly $E^{(n,m)} < 0$ for all $n > m$). Due to the forced anti-symmetry of the matrix there is redundancy of information, so the overall Spike Train Order can simply be derived as the sum over the upper right tridiagonal part of the matrix $E^{(n,m)}$:

$$E_{<} = \sum_{n < m} E^{(n,m)}. \quad (26)$$

Normalizing this cumulative quantity by the total number of possible coincidences yields the second definition of the Synfire Indicator:

$$F = \frac{2E_{<}}{(N-1)M}. \quad (27)$$

This definition is equivalent to Eq. 24. The only difference is that here the temporal summation over the profile is performed before and not after the spatial summation over spike train pairs.

The Synfire Indicator quantifies to what degree coinciding spike pairs with correct order prevail over coinciding spike pairs with incorrect order. It is normalized between -1 and 1 and the value 1 corresponds to a perfect synfire chain while the value -1 is obtained for a perfectly inverse synfire chain (i.e., one where the last spike train leads and the first spike train follows). It thus becomes clear that this quantity is a function of the spike train order (which can be denoted as $\varphi(n)$) and that maximizing the Synfire Indicator F_φ (starting from the initial (unsorted) order of the spike trains φ_u) finds the sorting of the spike trains from leader to follower such that the sorted set φ_s comes as close as possible to a perfect synfire pattern [12]:

$$\varphi_s : F_{\varphi_s} = \max_{\varphi} \{F_\varphi\} = F_s. \quad (28)$$

Unlike the unsorted Synfire Indicator $F_{\varphi_u} = F_u$, the optimized Synfire Indicator F_s can only attain values between 0 and 1 (for any negative value simply inverting the spike train order makes it positive, and that is even before the actual optimizing). However, from Eq. 22 it follows that for any given dataset F can never be higher than the SPIKE-Synchronization C (Eq. 19). The maximum value $F = 1$ is only attained when the spike train set can be sorted into a perfect synfire chain.

As can be appreciated in Fig. 6, according to its two alternative definitions, the maximization of the Synfire Indicator is expressed in both the normalized sum of the upper right half of the pairwise cumulative Spike Train Order matrix (Eq. 27, Fig. 6C) and in the average value of the Spike Train Order profile (Eq. 24, Fig. 6D). Along with this, the spike trains (Fig. 6E) are now sorted such that the first spike trains have predominantly high values (red) and the last spike trains predominantly low values (blue) of the SPIKE-Order.

Fig. 6 also illustrates that the results of the complete analysis contain several levels of information. Time-resolved (local) information is represented in the coloring of the spikes (according to the SPIKE-Order D) and in the profile of Spike Train Order E . Each element of the Spike Train Order matrix characterizes the leader-follower relationship between two spike trains at a time. The Synfire Indicator F characterizes the closeness of the whole dataset to a synfire pattern, both for the unsorted (F_u) and for

the sorted (F_s) spike trains. The sorted order of the spike trains is a very important result in itself since it identifies the leading and the following spike trains. Finally, as an important last step in the analysis it is highly recommended to evaluate the statistical significance of the optimized Synfire Indicator F_s using a set of carefully constructed spike train surrogates (not shown here but fully explained and visualized in [12]).

SPIKE-Order - Example applications

When originally proposed in [12], the Spike Train Order framework was applied to evaluate the consistency of the leader-follower relationships in datasets from two very different fields, neuroscience (giant depolarized potentials in mice slices) and climatology (El Niño sea surface temperature recordings). Later studies used the same framework to demonstrate the reproducibility of activation sequences in children with refractory epilepsy [39] and to perform a spatiotemporal analysis of cortical activity obtained by wide-field calcium images in mice before and after stroke [40].

4. Latency correction

4.1. Latency correction without overlap

When estimating synchrony within a spike train set, systemic delays such as the ones in the synfire chain of Fig. 5A are a hindrance, since usually the real question is how would the synchrony look like if there was no latency. For example, in the context of neuronal coding, before estimating the reliability of the neuronal responses upon repeated presentation of a stimulus, it would be best to first get rid of any variations in onset latency. Similarly, if the aim is to quantify the faithfulness in the propagation of activity from one neuron to another, this also should be done only after the removal of the propagation delays. The process of eliminating such systematic delays is called latency correction, and the "true" level of synchrony after such realigning of the spike trains is usually higher than the original synchrony. An algorithm for such a latency correction has recently been proposed in [13].

The crucial step is to go beyond SPIKE-Synchronization and Spike Train Order and to not only use their notions of coincidence and order but also take into account the actual temporal intervals

between matching spikes. Therefore, after spike matching via adaptive coincidence detection (Eqs. 15 and 16) the time difference between any matched pair of spikes is calculated as

$$\delta_i^{(n,m)} = t_i^{(n)} - t_{j'}^{(m)}, \quad (29)$$

where j' again identifies the coincident spike in spike-train m that matches spike i in spike train n according to Eq. 20.

Averaging over all the matched spikes for pair n, m of spike trains

$$\delta^{(n,m)} = \frac{1}{\sum_i C_i^{(n,m)}} \sum_i C_i^{(n,m)} \delta_i^{(n,m)} \quad (30)$$

yields the antisymmetric $N \times N$ *spike time difference matrix* (STDM) which estimates the pairwise latencies between all spike trains. Similarly, from the same pairwise averaged spike time differences a symmetric *cost matrix* can be defined as

$$c^{(n,m)} = \sqrt{\frac{1}{\sum_i C_i^{(n,m)}} \sum_i C_i^{(n,m)} [\delta_i^{(n,m)}]^2}, \quad (31)$$

which, in contrast to Eq. 30, guarantees that the value 0 is obtained if and only if all matched spike pairs are exactly coincident, i.e., all time differences are 0. The aim of latency correction then becomes to maximally align the spike trains by minimizing the *cost function* c , defined as the average of the upper right triangular part of the cost matrix:

$$c = \frac{2}{N(N-1)} \sum_{n < m}^N c^{(n,m)}. \quad (32)$$

In [13] two latency correction algorithms were proposed. The first algorithm, *direct shift*, is simple and fast since it takes into account only a minimal part ($N-1$ values) of the cost matrix. In the *Row Direct Shift* variant these are the values from one row (typically the first, which means that the first spike train is used as reference), while for the *First Diagonal Direct Shift* variant the first upper diagonal (the difference between neighboring spike trains) of the cost matrix is used to calculate the cumulative differences to the first spike train. The correction is performed by shifting the spike trains such that the corresponding matrix elements are set to 0 and the hope is that this way also the other $(N-1) * (N-2)/2$ matrix elements are minimized.

In Fig. 7 on the right hand side the STDM and the cost matrix are shown before and after the latency correction (done using the first row direct shift) for a regularly spaced synfire chain with a few missing and extra spikes as well as some jitter. For the uncorrected synfire chain (Fig. 7A), the further apart two spike trains, the larger the intervals between their matching spikes, and accordingly the values of the STDM and the cost matrix tend to increase with the distance from the diagonal (which itself is necessarily 0). After the latency correction (Fig. 7B) both matrices approach 0 everywhere reflecting a much improved alignment of the spike trains, but because of the remaining jitter (which can not be corrected) they do not go all the way down to 0.

Since in principle the search space of all possible shifts is infinite, the second latency correction algorithm proposed in [13] is based on *simulated annealing*, a heuristic approach which employs an iterative Monte Carlo algorithm to minimize the cost function and find the shifts that align the spike trains best. Starting from the initial spike train set and its original cost function c_{start} , in each iteration a randomly selected spike train is shifted by a randomly selected time interval (which decreases over time as the cost converges). The cost matrix is updated and the shift is accepted whenever this leads to a decrease of the cost. Unlike greedy algorithms, simulated annealing usually does not get stuck in local minima, since escape is possible because even shifts that lead to an increase in cost are allowed with a certain likelihood (which gets lower and lower as the temperature of the cooling scheme decreases). This iterative procedure continues until the cost function converges towards its final cost c_{end} .

For both algorithms the input is a set of N spike trains and there are two major outputs: the end cost c_{end} and the shifts $\vec{s} = [s_1; \dots; s_N]$ performed in order to get there. In [13] it was shown that simulated annealing in general achieves better results (e.g., lower end costs) than direct shifts, however, this improvement comes with a large computational cost of order N^2 (whereas for both direct shifts only $N-1$ additions or subtractions have to be performed).

4.2. Latency correction with overlap

The two algorithms introduced in Section 4.1 work very well as long as the global events are sufficiently

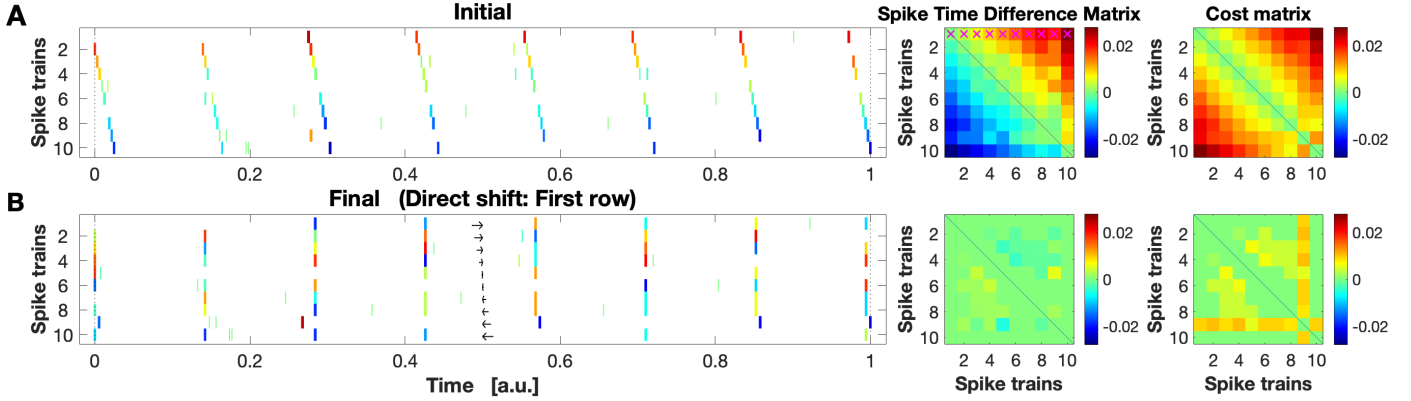


Figure 7: Latency correction performed via first row direct shift on a regularly spaced synfire chain with both missing and extra spikes as well as some jitter. A. Before and B. After the latency correction. The SPIKE-Order D is color-coded on a scale from 1 (red, first leader) to -1 (blue, last follower). In B the shift performed during the latency correction for each spike train is indicated by arrows. The spike time difference matrix (STDM) and the cost matrix turn from their rather ordered increase away from the diagonal (since initially spikes from more separated spike trains exhibit greater time separation) to very low values everywhere (because the corrected spike trains are almost perfectly aligned). The elements of the STDM used in the direct shift are marked by red crosses. Modified from [13].

separated, for example in the case of a perfect synfire chain such as the one shown in Fig. 8a, subplot A. Here each spike of every global event is coincident with all the other spikes of the same event and within each event all spikes are in perfect order. Accordingly, the pairwise matrices of both the SPIKE-Synchronization C (subplot B) and Spike Train Order E (subplot C) attain the maximum value of 1 everywhere and, in consequence, the same holds true for the overall SPIKE-Synchronization C (Eq. 19) and the Synfire Indicator F (Eq. 27). Finally, the cost matrix (subplot D, Eq. 31) exhibits a monotonous increase with distance from the main diagonal, as can be quantified by averaging the values of the different diagonals of the cost matrix (subplot E).

However, difficulties start to arise as soon as coincidence windows (Eq. 15) of neighboring events overlap, and these difficulties affect not only SPIKE-Synchronization, Spike Train Order and the Synfire Indicator but also the cost and therefore the latency correction as well. In Fig. 8b, subplot A, an example is shown in which the intervals between successive spikes from each global event have become so large that the last three spikes from each event are no longer coincident with the first spikes from the same event but rather with the first spikes from the next event. These spurious mismatches lead to diminishing SPIKE-Synchronization (subplot B) and to inconsistencies in the Spike Train Order (subplot C), and consequently SPIKE-Synchronization $C = 0.956$ and Synfire Indicator $F = 0.778$ are both lower than 1. Also the cost matrix (subplot D) no

longer monotonously increases and instead reaches its maximum not in the very last corner (as in Fig. 8a) but already at an intermediate diagonal (subplot E).

Under normal circumstances the criterion "closeness in time" used by the adaptive coincidence detection of Eqs. 15 and 16 to match spikes is perfectly reasonable. However, in the case of overlap it leads to the result that some of the spikes matched with each other actually belong to different events. But from the rasterplot one can often clearly recognize that the trailing spikes of an event should belong to that event and not to the next on (or sometimes this can be deduced from external knowledge that is not available to the impartial algorithm). In these cases the mismatches and all the resulting reductions in SPIKE-Synchronization, Synfire Indicator and spike time difference can be considered as spurious. In [14] the algorithms for latency correction from Section 4.1 were adapted such that in the case of overlapping global events they succeed to "overwrite" the naive event matching and still achieve the desired spike train alignment.

Both the row and the first diagonal direct shift algorithm introduced in Section 4.1 make use of only a very small part of the STDM matrix and are thus very sensitive to noise in the data: if just one of these $N - 1$ values is not reliable, the resulting shifts can be suboptimal and the overall error large. On the other hand, for simulated annealing the cost function is based on the whole STDM which in the case of overlap also includes the spurious outer parts of

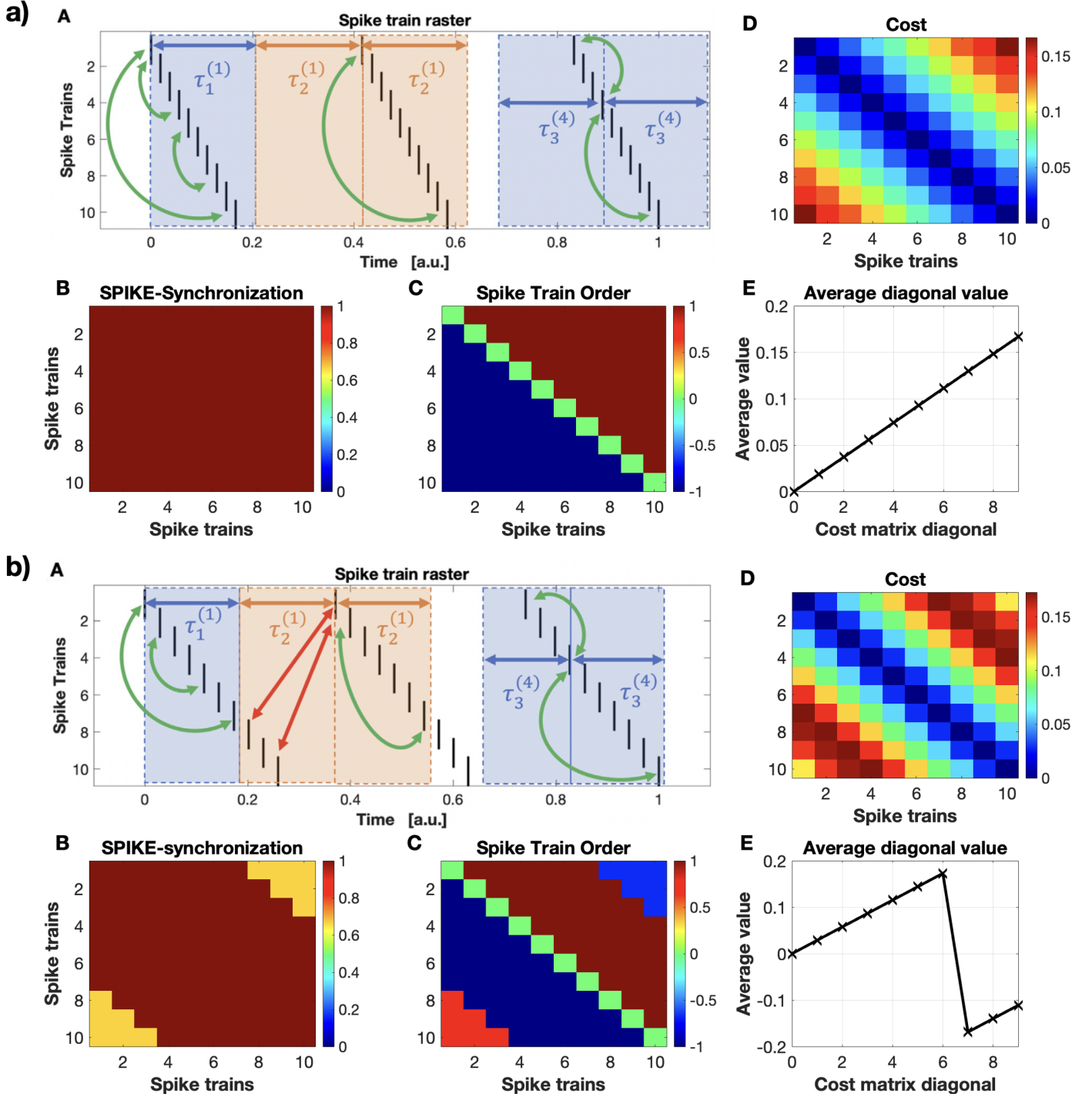


Figure 8: Illustrating the effect of event overlap using a perfect synfire chain with three global events without (subpanel a) and with (subpanel b) overlap. Both subpanels follow the same structure: A. Spike train sets. Coincidence windows of the i -th spike in the (n) -th spike train $\tau_i^{(n)}$ are indicated alternately in blue and orange. A few correct and spurious matches are marked by green (both subplots) and red (subplot b only) arrows, respectively. B. SPIKE-Synchronization and C. Spike Train Order matrix. Without overlap (a) both matrices attain maximal values, whereas in (b) overlap results in spurious matches which lead to deviations from 1 in the off diagonal corners. D. Cost matrix and E. Average diagonal value of the cost matrix. While without overlap (a) the cost increases monotonically with the distance from the diagonal, with overlap (b) the maximum is reached already at an intermediate diagonal. Modified from [14].

the matrix (see Fig. 8b) and this of course causes its own problems. Therefore, the solutions proposed in [14] address both of these problems by taking the good parts of each of the two existing approaches

while avoiding the drawbacks. First, the new fast direct shift algorithm *Extrapolation* takes into account a larger part of the spike time difference matrix than the two direct shifts (and is thus more robust with

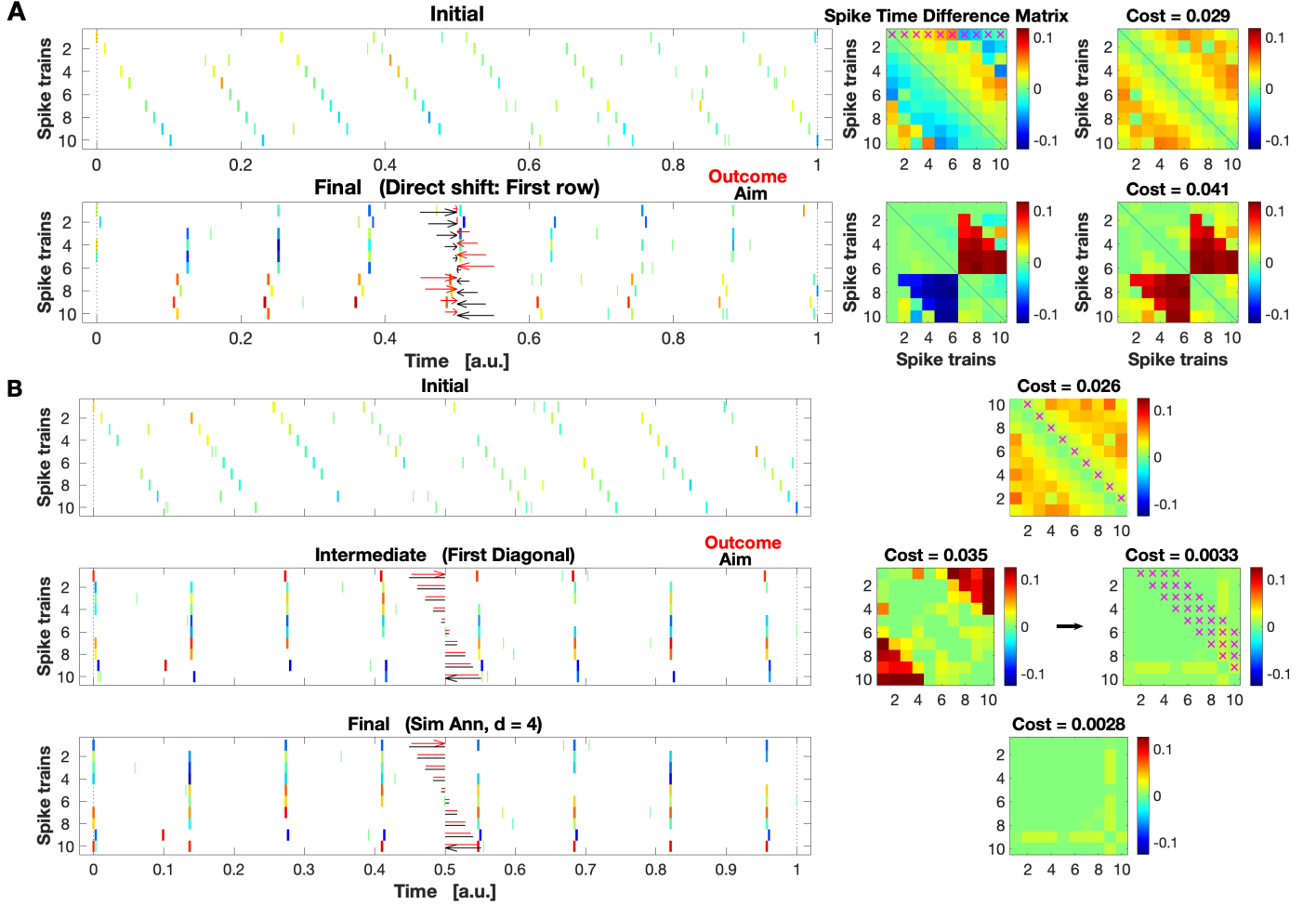


Figure 9: Iterative scheme improves on a simple direct shift when applied to a simulated synfire chain with overlapping incomplete events and some background spikes. A. The direct shift based on spike time difference values of the first row only (marked by red crosses in the spike time difference matrix) includes incorrect information from the off-diagonal corners of the STDM which are affected by spurious spike matches. It thus tries to align pairs of spikes that are actually mismatched and this erroneously breaks up one global event into two parts. B. Iterative scheme composed of a first diagonal direct shift (which avoids the spurious spike matches in the outer parts of the STDM) and a reduced matrix simulated annealing which aims to minimize the cost values up to the fourth diagonal only (again marked by red crosses). The first iteration actually increases the cost value since it is still based on spurious spike matches, but after the rematching (indicated by a black arrow on the right), there is a remarkable decrease in the cost value. In the second iteration the simulated annealing results in a further improvement. In both A and B the shifts that would eliminate the systematic delays of the synfire chain (Aim) and the shifts obtained by the latency correction algorithms (Outcome) are represented by black and red arrows, respectively. Modified from [14].

respect to noise), but at the same time manages to avoid the outer parts of the STDM that are affected by the spurious matches caused by overlap. Subsequently, also simulated annealing is modified such that is based on a reduced matrix only. Both algorithms use a parameter, the *stop diagonal* d , to determine the extent to which the STDM is used.

- Reduced Matrix Direct Shift: Extrapolation

The Extrapolation algorithm builds on a reduced part of the spike time difference matrix and uses the transitivity property to replace the overlap-affected outer parts of the STDM by extrapolat-

ing the unaffected inner parts of the matrix:

$$\delta^{(n,m)} = \sum_{k=m+1}^{n-1} [\delta^{(n,k)} + \delta^{(k,m)}] \quad (33)$$

for any n, m with $m - n > d$ where d is the stop diagonal parameter. To maintain the anti-symmetry of the STDM, the corresponding elements in the opposite half are updated accordingly:

$$\delta^{(m,n)} = -\delta^{(n,m)}. \quad (34)$$

Conveniently, since each row/column of the STDM contains the shift needed to align with the respective spike train (consistent with a main

diagonal of 0), a simple average over the full STDM immediately results in the invariant shift that uses the median spike train as reference:

$$s_n = \frac{1}{N} \sum_m \delta^{(m,n)}. \quad (35)$$

- **Reduced Matrix Simulated Annealing**

For simulated annealing the adaptation to overlapping global events is even more straightforward. Instead of extracting the cost function from the whole upper right triangular part of the STDM, now only the inner overlap-unaffected part of the STDM is utilized, that is, Eq. 32 in Section 4.1 is modified such that the sum skips all parts beyond the stop diagonal d :

$$c = \frac{2}{(N-d)(N-d-1)} \sum_{m=n>d}^N c^{(n,m)}. \quad (36)$$

These two algorithms that are both based on reduced matrices already achieve a much better latency correction for cases with overlap, but this can be pushed even further by means of an iterative scheme [14] in which each iteration consists of two steps: After an initial spike matching via adaptive coincidence detection a first latency correction is performed using a reduced matrix such that it relies only on spike pairs that are not affected by overlap. The aim is to disentangle the different global events and to eliminate as much overlap as possible. Then the second iteration starts again with an adaptive coincidence detection, this time based on spike pairs with a considerably improved alignment and is followed by another latency correction which now typically yields a very increased performance (i.e., lower cost).

This improvement is illustrated in Fig. 9 on slightly noisy synfire chains with overlapping events. The "old" first row direct shift of Section 4.1 (Fig. 9A) relies on the elements of the first row only and, even worse, incorporates spurious information from some of the overlap-affected values in the off-diagonal corner of the STDM, which together leads to an excessively high cost value. Instead, in the first iteration of the "new" iterative scheme (Fig. 9B) a first diagonal direct shift is used to eliminate the influence of the outer STDM elements that are affected by the spurious spike matches caused by overlap. Since at this moment the calculation of the cost

value itself still takes into account outer matrix elements that are based on spurious spike matches, this step actually leads to an increase in the cost (from 0.026 to 0.035). However, once the spikes that are now much better aligned within real global events are matched once more, the true cost value turns out to be very low (0.0033). The second iteration which employs reduced matrix simulated annealing with stop diagonal $d = 4$ decreases the cost value even further to a final value of 0.0028.

Finally, in a more extended study on artificially generated data [14] it could be shown that the Extrapolation direct shift achieves almost the same performance as the reduced matrix simulated annealing, even though it is much faster. For very large datasets this high computational cost of simulated annealing can actually be an exclusion criterion.

Latency Correction - Example applications

When it was proposed in [13], the regular latency correction algorithm of Section 4.1 (using simulated annealing) was applied to global activation patterns recorded via wide-field calcium imaging in the cortex of mice before and after stroke induced by means of a photothrombotic lesion (these data were first analyzed in [40]). In a comparison of three rehabilitation paradigms (motor training for healthy controls, pure motor training for mice with stroke, as well as mice with stroke and additional transient pharmacological inactivation of the contralesional hemisphere), the latter group which is the only one associated with general recovery was also the one that distinguished itself by its lowest end costs. On the other hand, in [14] the latency correction algorithm for data that contain global events with overlap was tested on single-unit recordings from two medial superior olive neurons of an anaesthetized gerbil during presentations of ten different noisy auditory stimuli [41]. In all cases the iterative scheme based on the fast extrapolation direct shift (see Section 4.2) managed to reduce the cost considerable.

5. Conclusion

Looking into the future, there are several promising avenues for methodological advancement: For the SPIKE-Distance and SPIKE-Synchronization (and potentially even SPIKE-Order) one could try to

find a way to add event magnitudes as weights to individual events, similar to what has been done with the earth mover distance (which contains the Victor-Purpura metric as a special case [42]). Among others, this could be very interesting in a meta-analysis of epileptic seizure recordings, where the length of the seizure would serve as weight. Comparing results with or without using these weights would provide very valuable information about the significance of the seizure duration, for example in epileptic seizure risk forecasting [43]. In latency correction there is still a need to explore in more detail the intertwined relationship between the sorting of spike trains (which is based on latencies as well) and the latency correction itself [14].

Last but not least, there are many more potential applications to experimental data: For the two spike train distances (ISI and SPIKE) and SPIKE-synchronization (as well as their adaptive generalizations [15]) these include tracking the neuronal reliability in datasets similar to the one in [1], more thorough investigations of stimulus discrimination and clustering in the context of neuronal coding [3], as well as detailed analysis of spatio-temporal propagation patterns in state of the art multi-electrode recordings [44]. On the other hand, the Spike Train Order framework combined with latency correction will allow to judge the faithfulness of activity propagation between different neurons or between different brain areas [45] or, in the context of neuronal coding, to estimate to what extent the response to repeated presentation of a stimulus is independent of variations in onset latency [46].

While all of these measures and algorithms have been developed within a neuroscientific context, the algorithms are universal and can be applied to discrete datasets in many scientific fields, from climatology (where the measured variable is typically either the temperature or the amount of rainfall) [12, 47, 48] via network science [49], social communication [50], and mobile communication [51] to policy diffusion [52].

Optimized implementations of all the measures of spike train synchrony and directionality are available in three free software packages. These are the Matlab graphical user interface SPIKY¹ [11], the Python

library PySpike² [53] and the Matlab command line library cSPIKE³. All of these will soon also include the various algorithms for latency correction as well as a recently proposed algorithm that finds within a larger neuronal population the most discriminative subpopulation [54].

References

- [1] Z. Mainen, T. J. Sejnowski, Reliability of spike timing in neocortical neurons, *Science* 268 (1995) 1503. doi:<https://doi.org/10.1126/science.7770778>.
- [2] D. Chicharro, T. Kreuz, R. G. Andrzejak, What can spike train distances tell us about the neural code?, *J Neurosci Methods* 199 (2011) 146–165. doi:<https://doi.org/10.1016/j.jneumeth.2011.05.002>.
- [3] R. Quian Quiroga, S. Panzeri, Principles of neural coding, CRC Taylor and Francis, Boca Raton, FL, USA, 2013. doi:<https://doi.org/10.1201/b14756>.
- [4] J. D. Victor, K. P. Purpura, Nature and precision of temporal coding in visual cortex: A metric-space analysis, *J Neurophysiol* 76 (1996) 1310. doi:<https://doi.org/10.1152/jn.1996.76.2.1310>.
- [5] M. C. W. van Rossum, A novel spike distance, *Neural Comput* 13 (2001) 751. doi:<https://doi.org/10.1162/089976601300014321>.
- [6] J. Haas, J. White, Frequency selectivity of layer ii stellate cells in the medial entorhinal cortex, *J. Neurophysiol.* 88 (2002) 2422. doi:<https://doi.org/10.1152/jn.00598.2002>.
- [7] S. Schreiber, J. M. Fellous, J. H. Whitmer, P. H. E. Tiesinga, T. J. Sejnowski, A new correlation-based measure of spike timing reliability, *Neurocomputing* 52 (2003) 925. doi:[https://doi.org/10.1016/S0925-2312\(02\)00838-X](https://doi.org/10.1016/S0925-2312(02)00838-X).
- [8] J. D. Hunter, G. Milton, Amplitude and frequency dependence of spike timing: implications for dynamic regulation, *J Neurophysiol* 90 (2003) 387. doi:<https://doi.org/10.1152/jn.00074.2003>.

¹<http://www.thomaskreuz.org/source-codes/SPIKY>

²<http://mariomulansky.github.io/PySpike>

³<http://www.thomaskreuz.org/source-codes/cSPIKE>

- [9] T. Kreuz, J. S. Haas, A. Morelli, H. D. I. Abarbanel, A. Politi, Measuring spike train synchrony, *J Neurosci Methods* 165 (2007) 151. doi:https://doi.org/10.1016/j.jneumeth.2007.05.031.
- [10] T. Kreuz, D. Chicharro, C. Houghton, R. G. Andrzejak, F. Mormann, Monitoring spike train synchrony, *J Neurophysiol* 109 (2013) 1457. doi:https://doi.org/10.1152/jn.00873.2012.
- [11] T. Kreuz, M. Mulansky, N. Bozanic, SPIKY: A graphical user interface for monitoring spike train synchrony, *J Neurophysiol* 113 (2015) 3432. doi:https://doi.org/10.1152/jn.00848.2014.
- [12] T. Kreuz, E. Satuavuori, M. Pofahl, M. Mulansky, Leaders and followers: Quantifying consistency in spatio-temporal propagation patterns, *New Journal of Physics* 19 (2017) 043028. doi:https://doi.org/10.1088/1367-2630/aa68c3.
- [13] T. Kreuz, F. Senocrate, G. Cecchini, C. Checcucci, A. L. A. Mascaro, E. Conti, A. Scaglione, F. S. Pavone, Latency correction in sparse neuronal spike trains, *J Neurosci Methods* 381 (2022) 109703. doi:https://doi.org/10.1016/j.jneumeth.2022.109703.
- [14] A. Mariani, F. Senocrate, J. Mikiel-Hunter, D. McAlpine, B. Beiderbeck, M. Pecka, K. Lin, T. Kreuz, Latency correction in sparse neuronal spike trains, *J Neurosci Methods* 416 (2025) 110378. doi:https://doi.org/10.1016/j.jneumeth.2025.110378.
- [15] E. Satuavuori, M. Mulansky, N. Bozanic, I. Malvestio, F. Zeldenrust, K. Lenk, T. Kreuz, Measures of spike train synchrony for data with multiple time scales, *J Neurosci Methods* 287 (2017) 25–38. doi:https://doi.org/10.1016/j.jneumeth.2017.05.028.
- [16] T. Kreuz, D. Chicharro, R. G. Andrzejak, J. S. Haas, H. D. I. Abarbanel, Measuring multiple spike train synchrony, *J Neurosci Methods* 183 (2009) 287. doi:https://doi.org/10.1016/j.jneumeth.2009.06.039.
- [17] E. Satuavuori, I. Malvestio, T. Kreuz, Measures of spike train synchrony and directionality, in: G. Naldi, T. Nieuws (Eds.), *Mathematical and Theoretical Neuroscience*, Vol. 24 of Springer INdAM Series, Springer Cham, Berlin, 2017, p. 201. doi:https://doi.org/10.1007/978-3-319-68297-6.
- [18] E. Satuavuori, T. Kreuz, Which spike train distance is most suitable for distinguishing rate and temporal coding?, *J Neurosci Methods* 299 (2018) 22. doi:https://doi.org/10.1016/j.jneumeth.2018.02.009.
- [19] W. M. Usrey, R. C. Reid, Synchronous activity in the visual system., *Annu. Rev. Physiol.* 61 (1999) 435. doi:https://dx.doi.org/10.1146/annurev.physiol.61.1.435.
- [20] P. H. E. Tiesinga, J. M. Fellous, T. J. Sejnowski, Regulation of spike timing in visual cortical circuits, *Nature Reviews Neuroscience* 9 (2008) 97. doi:https://doi.org/10.1038/nrn2315.
- [21] J. Shlens, F. Rieke, E. L. Chichilnisky, Synchronized firing in the retina, *Curr Opin Neurobiol* 18 (2008) 396. doi:https://doi.org/10.1016/j.conb.2008.09.010.
- [22] T. Kreuz, D. Chicharro, M. Greschner, R. G. Andrzejak, Time-resolved and time-scale adaptive measures of spike train synchrony, *J Neurosci Methods* 195 (2011) 92. doi:https://doi.org/10.1016/j.jneumeth.2010.11.020.
- [23] M. Mulansky, N. Bozanic, A. Sburlea, T. Kreuz, A guide to time-resolved and parameter-free measures of spike train synchrony, 2015 International Conference on Event-based Control, Communication, and Signal Processing (EBCCSP) (2015) 1–8doi:https://doi.org/10.1109/EBCCSP.2015.7300693.
- [24] N. Doorn, M. J. van Putten, M. Frega, Automated inference of disease mechanisms in patient-hipsc-derived neuronal networks, *Communications Biology* 8 (1) (2025) 768. doi:https://doi.org/10.1038/s42003-025-08209-2.
- [25] F. Moradi, M. van den Berg, M. Mirjebreili, L. Kosten, M. Verhoye, M. Amiri, G. A. Keliris, Early classification of alzheimer’s disease phenotype based on hippocampal electrophysiology in the

- tgf344-ad rat model, *iScience* 26 (8) (2023). doi:<https://doi.org/10.1016/j.isci.2023.107454>.
- [26] N. Melanitis, K. S. Nikita, Biologically-inspired image processing in computational retina models, *Computers in Biology and Medicine* 113 (2019) 103399. doi:<https://doi.org/10.1016/j.compbiomed.2019.103399>.
- [27] N. Lama, A. Hargreaves, B. Stevens, T. M. McGinnity, Spike train synchrony analysis of neuronal cultures, in: 2018 International Joint Conference on Neural Networks (IJCNN), IEEE, 2018, pp. 1–8. doi:<https://doi.org/10.1109/IJCNN.2018.8489728>.
- [28] L.-H. N. Lee, C. Y. Ngan, C.-K. Yang, R.-W. Wang, H.-J. Lai, C.-H. Chen, Y.-C. Yang, C.-C. Kuo, Motor cortex stimulation ameliorates parkinsonian locomotor deficits: effectual and mechanistic differences from subthalamic modulation, *npj Parkinson's Disease* 11 (1) (2025) 32. doi:<https://doi.org/10.1038/s41531-025-00879-3>.
- [29] N. Goshi, D. Lam, C. Bogguri, V. K. George, A. Sebastian, J. Cadena, N. F. Leon, N. R. Hum, D. R. Weilhammer, N. O. Fischer, et al., Direct effects of prolonged $\text{tnf-}\alpha$ and il-6 exposure on neural activity in human ipsc-derived neuron-astrocyte co-cultures, *Frontiers in Cellular Neuroscience* 19 (2025) 1512591. doi:<https://doi.org/10.3389/fncel.2025.1512591>.
- [30] J. Jouty, G. Hilgen, E. Sernagor, M. H. Hennig, Non-parametric physiological classification of retinal ganglion cells in the mouse retina, *Frontiers in Cellular Neuroscience* 12 (2018) 481. doi:<https://doi.org/10.3389/fncel.2018.00481>.
- [31] R. Nishizono, N. Saijo, M. Kashino, Highly reproducible eyeblink timing during formula car driving, *iScience* 26 (6) (2023). doi:<https://doi.org/10.1016/j.isci.2023.106803>.
- [32] D. W. Gauthier, N. James, B. D. Auerbach, Altered auditory feature discrimination in a rat model of fragile x syndrome, *PLoS Biology* 23 (7) (2025) e3003248. doi:<https://doi.org/10.1101/2025.02.18.638956>.
- [33] J. C. Nocon, H. J. Gritton, N. M. James, R. A. Mount, Z. Qu, X. Han, K. Sen, Parvalbumin neurons enhance temporal coding and reduce cortical noise in complex auditory scenes, *Communications Biology* 6 (1) (2023) 751. doi:<https://doi.org/10.1038/s42003-023-05126-0>.
- [34] R. Quian Quiroga, T. Kreuz, P. Grassberger, Event synchronization: A simple and fast method to measure synchronicity and time delay patterns, *Phys Rev E* 66 (2002) 041904. doi:<https://doi.org/10.1103/PhysRevE.66.041904>.
- [35] K. Shan, C. Tian, Z. Zheng, M. Zheng, K. Xu, Short-term plasticity promotes synchronization of coupled chaotic oscillators in excitatory–inhibitory networks, *Chaos: An Interdisciplinary Journal of Nonlinear Science* 35 (7) (2025). doi:<https://doi.org/10.1063/5.0243837>.
- [36] H. Rostro-Gonzalez, E. I. Guerra-Hernandez, P. Batres-Mendoza, A. A. Garcia-Granada, M. Cano-Lara, A. Espinal, Enhancing legged robot locomotion through smooth transitions using spiking central pattern generators, *Biomimetics* 10 (6) (2025) 381. doi:<https://doi.org/10.3390/biomimetics10060381>.
- [37] X. Bai, C. Yu, J. Zhai, Topological data analysis of the firings of a network of stochastic spiking neurons, *Frontiers in Neural Circuits* 17 (2024) 1308629. doi:<https://doi.org/10.3389/fncir.2023.1308629>.
- [38] M. Seifert, P. A. Roberts, G. Kafetzis, D. Osorio, T. Baden, Birds multiplex spectral and temporal visual information via retinal on-and off-channels, *Nature Communications* 14 (1) (2023) 5308. doi:<https://doi.org/10.1038/s41467-023-41032-z>.
- [39] S. B. Tomlinson, J. N. Wong, E. C. Conrad, B. C. Kennedy, E. D. Marsh, Reproducibility of interictal spike propagation in children with refractory epilepsy, *Epilepsia* 60 (5) (2019) 898–910. doi:<https://doi.org/10.1111/epi.14720>.
- [40] G. Cecchini, A. Scaglione, A. L. Allegra Mascaro, C. Checcucci, E. Conti, I. Adam, D. Fanelli, R. Livi, F. S. Pavone,

- T. Kreuz, Cortical propagation tracks functional recovery after stroke, *PLoS Computational Biology* 17 (5) (2021) e1008963. doi:<https://doi.org/10.1371/journal.pcbi.1008963>.
- [41] B. Beiderbeck, Reading between the lines of the duplex theory, Ph.D. thesis, Graduate School of Systemic Neurosciences, Ludwig-Maximilians-Universität München (LMU), Germany (2022). doi:<https://doi.org/10.5282/edoc.30131>.
- [42] D. Sihm, S.-P. Kim, A spike train distance robust to firing rate changes based on the earth mover's distance, *Frontiers in Computational Neuroscience* 13 (2019) 82. doi:<https://doi.org/10.3389/fncom.2019.00082>.
- [43] M. G. Leguia, R. G. Andrzejak, C. Rummel, J. M. Fan, E. A. Mirro, T. K. Tcheng, V. R. Rao, M. O. Baud, Seizure cycles in focal epilepsy, *JAMA Neurology* 78 (4) (2021) 454–463. doi:<https://doi.org/10.1001/jamaneurol.2020.5370>.
- [44] M. Schröter, F. Cardes, C.-V. H. Bui, L. D. Dodi, T. Günswein, J. Bartram, L. Sadi-raj, P. Hornauer, S. Kumar, M. Pascual-Garcia, et al., Advances in large-scale electrophysiology with high-density micro-electrode arrays, *Lab on a Chip* (2025). doi:<https://doi.org/10.1039/d5lc00058k>.
- [45] A. Kumar, S. Rotter, A. Aertsen, Spiking activity propagation in neuronal networks: reconciling different perspectives on neural coding, *Nature Rev Neurosci* 11 (2010) 615–627. doi:<https://doi.org/10.1038/nrn2886>.
- [46] M. Levakova, M. Tamborrino, S. Ditlevsen, P. Lansky, A review of the methods for neuronal response latency estimation, *Biosystems* 136 (2015) 23–34. doi:<https://doi.org/10.1016/j.biosystems.2015.04.008>.
- [47] A. Y. Sun, Y. Xia, T. G. Caldwell, Z. Hao, Patterns of precipitation and soil moisture extremes in Texas, US: A complex network analysis, *Advances in Water Resources* 112 (2018) 203–213. doi:<https://doi.org/10.1016/j.advwatres.2017.12.019>.
- [48] F. R. Conticello, F. Cioffi, U. Lall, B. Merz, Synchronization and delay between circulation patterns and high streamflow events in Germany, *Water Resources Research* 56 (4) (2020) e2019WR025598. doi:<https://doi.org/10.1029/2019WR025598>.
- [49] V. Mwaffo, J. Keshavan, T. L. Hedrick, S. Humbert, Detecting intermittent switching leadership in coupled dynamical systems, *Scientific reports* 8 (1) (2018) 10338. doi:<https://doi.org/10.1038/s41598-018-28285-1>.
- [50] G. Varni, G. Volpe, A. Camurri, A system for real-time multimodal analysis of nonverbal affective social interaction in user-centric media, *IEEE Transactions on Multimedia* 12 (2010) 576. doi:<https://doi.org/10.1109/TMM.2010.2052592>.
- [51] L. Wang, G. Tan, C. Zang, Identifying the spatiotemporal organization of high-traffic events in a mobile communication system using event synchronization and complex networks, *Chaos: An Interdisciplinary Journal of Nonlinear Science* 32 (9) (2022). doi:<https://doi.org/10.1063/5.0083137>.
- [52] C. Grabow, J. Macinko, D. Silver, M. Porfiri, Detecting causality in policy diffusion processes, *Chaos: An Interdisciplinary Journal of Nonlinear Science* 26 (8) (2016). doi:<https://doi.org/10.1063/1.4961067>.
- [53] M. Mulansky, T. Kreuz, Pyspike - A python library for analyzing spike train synchrony, *Software X* 5 (2016) 183. doi:<https://doi.org/10.1016/j.softx.2016.07.006>.
- [54] E. Satuavuori, M. Mulansky, A. Daffertshoder, T. Kreuz, Using spike train distances to identify the most discriminative neuronal subpopulation, *J Neurosci Methods* 308 (2018) 354. doi:<https://doi.org/10.1016/j.jneumeth.2018.09.008>.