SPIKY: A graphical user interface for monitoring spike train synchrony

Thomas Kreuz, Nebojsa Bozanic

Institute for complex systems, CNR, Sesto Fiorentino, Italy

Abstract

Recently, the SPIKE-distance has been proposed as a measure of spike train synchrony which is both parameter-free and time-scale independent. Since it relies on instantaneous estimates of spike train dissimilarity, it is also time-resolved which makes it possible to track changes in instantaneous clustering, i.e., time-localized patterns of (dis)similarity among multiple spike trains. Further features include selective and triggered temporal averaging as well as the instantaneous comparison of spike train groups. Besides the regular SPIKE-distance, there also exists a causal variant which is defined such that the instantaneous values of dissimilarity rely on past information only so that time-resolved spike train synchrony can be estimated in real-time. Finally, here we introduce the future SPIKE-distance which can be used in triggered temporal averaging in order to evaluate the effect of certain spikes or of certain stimuli features on future spiking. In the first part of this report we address some of the computational aspects in the calculation and implementation of the SPIKE-distance while in the second part we present SPIKY, a graphical user interface which facilitates the application of the SPIKE-distance and all its variants to both simulated and real data.

1. Introduction

A wide variety of approaches to quantify the dissimilarity, or distance, between two spike trains has been suggested. Among these is the metric introduced in Victor and Purpura (1996), which evaluates the cost needed to transform one spike train into the other, using only certain elementary steps. Another metric proposed in van Rossum (2001), measures the Euclidean distance between the two spike trains after convolution of the spikes with an exponential function. Both methods involve one parameter that sets the time scale. In contrast, two more recent bivariate approaches, the ISI- and the SPIKE-distance are time scale independent and self-adaptive (Kreuz et al., 2007, 2013). These two measures are complementary: The ISI-distance relies on the relative length of simultaneous interspike intervals and is thus well-designed to quantify similarities in the neurons firing-rate profiles (Kreuz et al., 2007, 2009). The SPIKE-distance is based on differences between the spike times of the two spike trains and is therefore ideally suited to track synchrony that is mediated by spike timing (Kreuz et al., 2011, 2013).

XXXXX Computational aspects XXXXX

2. Methods

2.1. *The SPIKE-distance*

The SPIKE-distance (see Kreuz et al. (2011) for the original proposal and Kreuz et al. (2013) for the definite version presented here) is extracted from differences between the spike times of the two spike trains. It relies on instantaneous values in the sense that in a first step the two sequences of discrete spike times are transformed into a continuous dissimilarity profiles $S(t)$. This dissimilarity profile is based on three piecewise constant quantities which for each neuron $n = 1, 2$ are assigned to every time instant between 0 and T (see Fig. 1). These are the time of the preceding spike

$$t_{\mathrm{P}}^x(t) = \max(t_i^n | t_i^n \leq t) \quad t_1^n \leq t \leq t_{M_n}^n, \qquad (1)$$

the time of the following spike

$$t_{\mathrm{F}}^n(t) = \min(t_i^n | t_i^n > t) \quad t_1^n \leq t \leq t_{M_n}^n, \qquad (2)$$

as well as the interspike interval

$$x_{\mathrm{ISI}}^n(t) = t_{\mathrm{F}}^n(t) - t_{\mathrm{P}}^n(t). \qquad (3)$$

The ambiguity regarding the definition of the very first and the very last interspike interval is resolved by placing for each spike train auxiliary leading spikes at time $t = 0$ and auxiliary trailing spikes at time $t = T$. From these three quantities the dissimilarity profile is calculated in two steps: First for each spike the distance to the nearest spike in the other spike train is calculated, then for each time instant the

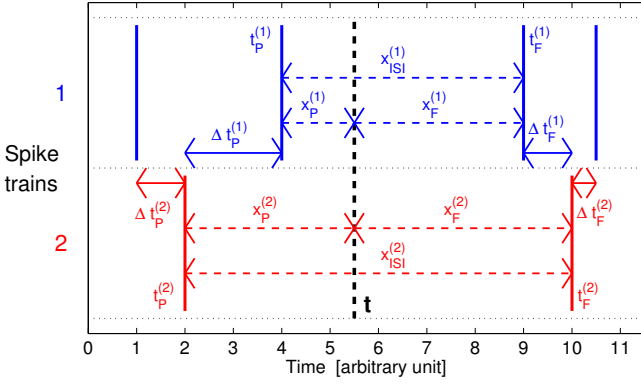---
Email address: `thomas.kreuz@cnr.it` (Thomas Kreuz).

Fig. 1. SPIKE-distance. Illustration of the local quantities needed to define the dissimilarity profile $S(t)$ for an arbitrary time instant t.

relevant spike time differences are selected, weighted, and normalized. Here relevant means local; each time instant is uniquely surrounded by four corner spikes: the preceding spike from the first spike train $t_P^{(1)}$, the following spike from the first spike train $t_F^{(1)}$, the preceding spike from the second spike train $t_P^{(2)}$, and, finally, the following spike from the second spike train $t_F^{(2)}$. Each of these corner spikes can be identified with a spike time difference, for example, for the previous spike of the first spike train

$$\Delta t_P^{(1)} = \min_i(|t_P^{(1)} - t_i^{(2)}|) \qquad (4)$$

and analogously for $t_F^{(1)}$, $t_P^{(2)}$, and $t_F^{(2)}$ (see Fig. 1). For each spike train separately a locally weighted average is employed such that the differences for the closer spike dominate; the weighting factors depend on

$$x_P^{(n)}(t) = t - t_P^{(n)}(t) \qquad (5)$$

and

$$x_F^{(n)}(t) = t_F^{(n)}(t) - t, \qquad (6)$$

the intervals to the previous and the following spikes for each neuron $n = 1, 2$. The local weighting for the spike time differences of the first spike train reads

$$S_1(t) = \frac{\Delta t_P^{(1)} x_F^{(1)} + \Delta t_F^{(1)} x_P^{(1)}}{x_{ISI}^{(1)}} \qquad (7)$$

and analogously $S_2(t)$ is obtained for the second spike train. Averaging over the two spike train contributions and normalizing by the mean interspike interval yields

$$S''(t) = \frac{S_1(t) + S_2(t)}{2\langle x_{ISI}^{(n)} \rangle_n}. \qquad (8)$$

This quantity weights the spike time differences for each spike train according to the relative distance of the corner spike from the time instant under investigation. This way relative distances within each spike train are taken care of, while relative distances between spike trains are not. In order to get these ratios straight and to account for differences in firing rate, in a last step the two contributions from the two spike trains are locally weighted by their instantaneous

interspike intervals. This leads to the improved definition of the dissimilarity profile

$$S(t) = \frac{S_1(t) x_{ISI}^{(2)} + S_2(t) x_{ISI}^{(1)}}{2\langle x_{ISI}^{(n)} \rangle_n^2}. \qquad (9)$$

The SPIKE-distance is defined as the temporal average of this dissimilarity profile

$$D_S = \frac{1}{T} \int_{t=0}^{T} dt\, S(t). \qquad (10)$$

The dissimilarity profile $S(t)$ and the SPIKE-distance D_S as its average are bounded in the interval $[0, 1]$. The distance value $D_S = 0$ is obtained for identical spike trains only.

There exists a straightforward extension to the case of more than two spike trains (number of spike trains $N > 2$), the averaged bivariate distance. This average over all pairs of neurons commutes with the average over time, so it is possible to achieve the same kind of time-resolved visualization as in the bivariate case by first calculating the instantaneous average, e.g., $S^a(t)$ over all pairwise instantaneous values $S^{mn}(t)$,

$$S^a(t) = \frac{1}{N(N-1)/2} \sum_{n=1}^{N-1} \sum_{m=n+1}^{N} S^{mn}(t) \qquad (11)$$

The Matlab source code for calculating and visualizing the SPIKE-distance can be found under http://www.fi.isc.cnr.it/users/thomas.kreuz/sourcecode.html.

xxx We define the ISI as going from right after the last spike to the exact time of the next spike. So the previous spike is not part of the ISI while the following is. So when you are on a spike this is the following spike and the one before is the previous spike.

2.2. Realtime SPIKE-distance

The Realtime SPIKE-distance D_{S_r} is a modification of the SPIKE-distance with the key difference that the corresponding time profile $S_r(t)$ can be calculated online because it relies on past information only. From the perspective of an online measure, the information provided by the following spikes, both their position and the length of the interspike interval, is not yet available. Like the regular (improved) SPIKE-distance D_S, this causal variant is also based on local spike time differences but now only two corner spikes are available, and the spikes of comparison are restricted to past spikes, e.g., for the preceding spike of the first spike train

$$\Delta t_P^{(1)} = \min_i(|t_P^{(1)} - t_i^{(2)}|), t_i < t. \qquad (12)$$

Since there are no following spikes available, there is no local weighting, and since there is no interspike interval, the normalization is achieved by dividing the average corner spike difference by twice the average time interval to the preceding spikes (Eq. 5). This yields a causal indicator of local spike train dissimilarity:

$$S_r(t) = \frac{\Delta t_{\mathrm{P}}^{(1)} + \Delta t_{\mathrm{P}}^{(2)}}{4\langle x_{\mathrm{P}}^{(n)}\rangle_n}. \qquad (13)$$

2.3. *Future SPIKE-distance*

The future SPIKE-distance D_{S_f} can be used in triggered temporal averaging in order to evaluate the effect of certain spikes or of certain stimuli features on future spiking. It is the inverse measure to the Realtime SPIKE-distance but instead of relying on past information only it relies on future information only. Also in this case for each time instant just two corner spikes are available, since the spikes of comparison are restricted to future spikes only. Thus the spike time difference for the following spike of the first spike train reads

$$\Delta t_{\mathrm{F}}^{(1)} = \min_i(|t_{\mathrm{F}}^{(1)} - t_i^{(2)}|), t_i > t, \qquad (14)$$

and accordingly for the following spike of the second spike train. And, similar to Eq. 13, an indicator of local spike train dissimilarity is obtained as follows:

$$S_f(t) = \frac{\Delta t_{\mathrm{F}}^{(1)} + \Delta t_{\mathrm{F}}^{(2)}}{4\langle x_{\mathrm{F}}^{(n)}\rangle_n}. \qquad (15)$$

2.4. *Edge effect*

suggested by Conor Houghton

2.5. *The ISI-distance*

While the dissimilarity profile of the SPIKE-distance is extracted from differences between the spike times of the two spike trains, the dissimilarity profile of the ISI-distance (Kreuz et al., 2007, 2009) is calculated as the instantaneous ratio between the interspike intervals $x_{\mathrm{ISI}}^{(1)}$ and $x_{\mathrm{ISI}}^{(2)}$ (Eq. 3) according to:

$$I(t) = \begin{cases} x_{\mathrm{ISI}}^{(1)}(t)/x_{\mathrm{ISI}}^{(2)}(t) - 1 & \text{if } x_{\mathrm{ISI}}^{(1)}(t) \le x_{\mathrm{ISI}}^{(2)}(t) \\ -(x_{\mathrm{ISI}}^{(2)}(t)/x_{\mathrm{ISI}}^{(1)}(t) - 1) & \text{otherwise.} \end{cases}$$
$$(16)$$

This ISI-ratio equals 0 for identical ISI in the two spike trains, and approaches -1 and 1, respectively, if the first or the second spike train is much faster than the other. For the ISI-distance the temporal averaging analogous to Eq. is performed on the absolute value of the ISI-ratio, thus both kinds of deviations are treated equally. Since the ISI-distance relies on the instantaneous ISI-values and thus requires knowledge about the following spikes, no causal Realtime extension is possible.

2.6. *Representations*

The ISI- and the SPIKE-distance combine a variety of properties that make them well suited for applications to real data. In particular, they are conceptually simple,

computationally efficient, and easy to visualize in a time-resolved manner. By taking into account only the preceding and the following spike in each spike train, these distances rely on local information only. They are also time-scale-adaptive since the information used is not contained within a window of fixed size but rather within a time frame whose size depends on the local rate of each spike train.

Moreover, the sensitivity to spike timing and the instantaneous reliability achieved by the SPIKE-distance opens up many new possibilities in multi-neuron spike train analysis (Kreuz et al., 2013). These build upon the fact that there are several levels of information reduction.

2.6.1. *Full matrix and cross sections*
The starting point is the most detailed representation in which one instantaneous value is obtained for each pair of spike trains (see Eq. 9). For multineuron data, this results in a matrix of size number of sampled time instants s squared number of spike trains (i.e., $\#(t_s)N^2$).

From this matrix, it is possible to extract any desired information. By selecting a pair of spike trains, one obtains the bivariate dissimilarity profile S(t) for this pair of spike trains. Selecting a time instant t_s instead yields an instantaneous matrix of pairwise spike train dissimilarities $S_{mn}(t_s)$. This matrix can be used to divide the spike trains into instantaneous clusters, that is, groups of spike trains with low intragroup and high intergroup dissimilarity.

2.6.2. *Spatial and temporal averaging*
Another way to reduce the information of the dissimilarity matrix is averaging. There are two possibilities that commute: the spatial average over spike train pairs and the temporal average.

The local average over spike train pairs yields a dissimilarity profile for the whole population. Temporal averaging over certain (continuous or noncontinuous) intervals on the other hand leads to a bivariate distance matrix (in real data, these intervals could be chosen to correspond to different external conditions such as normal vs. pathological, asleep vs. awake, target vs. nontarget stimulus, or presence/absence of a certain channel blocker). Finally, in both cases, application of the respective remaining average results in one distance value that describes the overall level of synchrony for a group of spike trains over a given time interval.

2.6.3. *Triggered averaging*
The fact that there are no limits to the temporal resolution allows further analyses such as internally or externally triggered temporal averaging. Here, the matrices are averaged over certain trigger time instants only. The idea is to check whether this triggered temporal average is significantly different from the global average since this would indicate that something peculiar is happening at these trigger instants. These trigger times can either be obtained from internal conditions (such as the spike times of a cer-
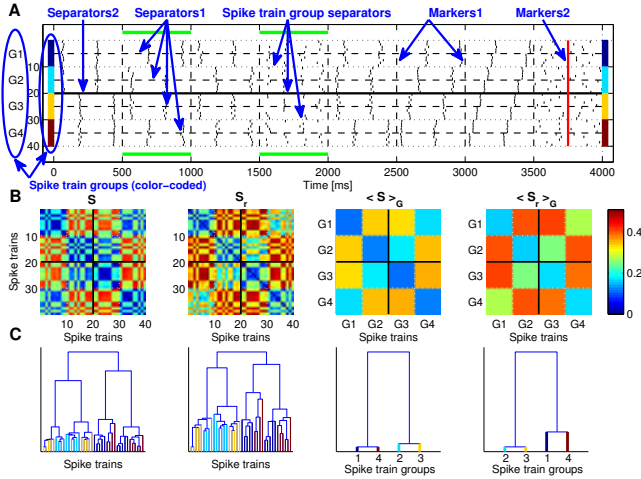
Fig. 2. Annotated screenshot from a movie. A. Artificially generated spike trains. B. Dissimilarity matrices obtained by averaging over two separate time intervals for both the regular and the real-time SPIKE-distance as well as their averages over subgroups of spike trains (denoted by $< >_G$). C. Corresponding dendrograms.

tain spike train) or from external influences (such as the occurrence of certain features in a stimulus). In real multi-neuron data, internal triggering might help to uncover the connectivity in neural networks or to detect converging or diverging patterns of firing propagation. External triggering might be helpful in addressing questions of neuronal coding, for example, it could be used to evaluate the influence of localized stimulus features on the reliability of real neurons under repeated stimulation.

A last possibility is spatial averaging such that the spike trains are manually assigned to subgroups, and a block matrix (and the corresponding dendrogram) is obtained by averaging over the respective submatrices of the original dissimilarity matrix. In applications to real data, these groups could be different neuronal populations or responses to different stimuli, depending on whether the spike trains were recorded simultaneously or successively.

3. Results

3.1. *Computational aspects*

XXXXX Rewrite and extend, add part on MEX-files XXXXX

The improved dissimilarity profile $S(t)$ of the SPIKE-distance is piecewise linear (with each linear interval running from one spike of the pooled spike train to the next) rather than piecewise constant as is the case for the ISI-distance. Therefore, when the localized visualization is desired, a new value has to be calculated for each sampling point and not just once per each interval in the pooled spike train. In cases where the distance value itself is sufficient, the short computation time can be even further decreased by representing each interval by the value of its center and weighting it by its length. This is not only faster,
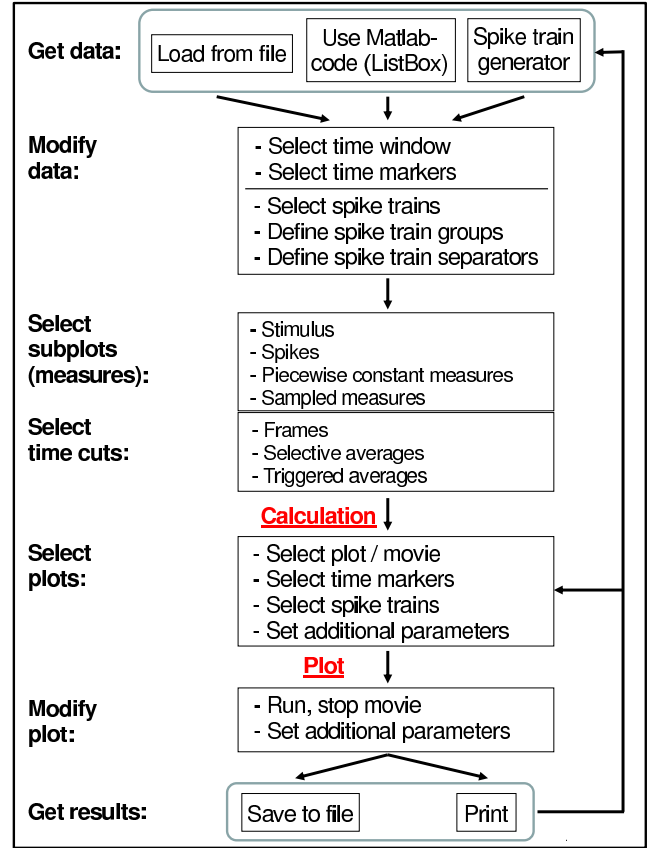

Fig. 3. Annotated screenshot from a movie. A. Artificially generated spike trains. B. Dissimilarity matrices obtained by averaging over two separate time intervals for both the regular and the real-time SPIKE-distance as well as their averages over subgroups of spike trains (denoted by $< >_G$). C. Corresponding dendrograms.

but it actually gives the exact result, whereas the time-resolved calculation is a very good approximation only for sufficiently small sampling intervals dt (imagine the example of a rectangular function, at some point any sampled representation has to cut the right angle). The dissimilarity profiles $S_r(t)$ and $S_f(t)$ of the real-time and the future SPIKE-distances are hyperbolic and not linear but here also the exact result can be obtained by piecewise integration over all intervals of the pooled spike train.

XXXXX Add figure on Pico vs. sampled XXXXX

The calculation of the SPIKE-distance consists of three steps: In a precalculation step, for each spike the distance to the nearest spike in all the other spike trains is calculated. Successively, for each time instant and each pair of spike trains, the distances of the four corner spikes are first locally weighted and then normalized. These latter steps involve matrices of the order number of time instants number of spike train pairs, which for very long datasets with many spike trains can lead to memory problems. The solution to these problems is to make the calculation sequential, i.e., to cut the recording interval into smaller segments, and to perform the averaging over all pairs of spike trains for each segment separately (no additional auxiliary spikes are needed except for very huge datasets for which even the cal-

4

culation of the first matrix is too memory-demanding). In the end the dissimilarity profiles for the different segments (already averaged over pairs of spike trains) are concatenated, and its temporal average yields the distance value for the whole recording interval.

The computational load scales with the number of spike trains N as N^2. More information on the implementation as well as the Matlab source code for both the ISI- and the SPIKE-distance (including its real-time variant) can be found at www.fi.isc.cnr.it/users/thomas.kreuz/sourcecode.html

3.2. *SPIKY*

3.3. *Spike train surrogates*

Daniel's Email 13.07.2011:

Regarding the use of Poisson processes to have a benchmark it will have different implications depending on the structure of your data. If the rate is almost constant then comparing to processes with the same estimated average rate is the best option. If the rate is time-dependent you may want to compare with processes which are Poissonian but mimic the same time-dependent profile of the rate. These are just alternatives that give you complementary information on the structure of the reliability.

Surrogate data with the same time dependent rate can be obtained by randomly reassigning each spike to any of the spike trains.

3.4. *Comparison with other implementations*

XXXXXXXXXX
Python-Implementation of SPIKE-distance courtesy of Jeremy Fix:
Python-Implementation of the pairwise ISI-distance courtesy of Michael Chary:
Răzvan Florian:
(Rusu and Florian, 2014)
Nebojša Božanić

4. Discussion

References

Kreuz, T., Chicharro, D., Andrzejak, R. G., Haas, J. S., Abarbanel, H. D. I., 2009. Measuring multiple spike train synchrony. J Neurosci Methods 183, 287.

Kreuz, T., Chicharro, D., Greschner, M., Andrzejak, R. G., 2011. Time-resolved and time-scale adaptive measures of spike train synchrony. J Neurosci Methods 195, 92.

Kreuz, T., Chicharro, D., Houghton, C., Andrzejak, R. G., Mormann, F., 2013. Monitoring spike train synchrony. J Neurophysiology 109, 1457.

Kreuz, T., Haas, J. S., Morelli, A., Abarbanel, H. D. I., Politi, A., 2007. Measuring spike train synchrony. J Neurosci Methods 165, 151.

Rusu, C. V., Florian, R. V., 2014. A new class of metrics for spike trains. Neural Computation 26 (2), 306–348.

van Rossum, M. C. W., 2001. A novel spike distance. Neural Computation 13, 751.

Victor, J. D., Purpura, K. P., 1996. Nature and precision of temporal coding in visual cortex: A metric-space analysis. J Neurophysiol 76, 1310.