Journal of Neuroscience Methods 190 (2010) 129–142

Contents lists available at ScienceDirect

Journal of Neuroscience Methods

journal homepage: www.elsevier.com/locate/jneumeth

NASS: An empirical approach to spike sorting with overlap resolution based on a hybrid noise-assisted methodology

Dimitrios A. Adamos[a,*], Nikolaos A. Laskaris[b], Efstratios K. Kosmidis[c], George Theophilidis[a]

[a] Laboratory of Animal Physiology, School of Biology, Aristotle University of Thessaloniki, 54 124 Thessaloniki, Greece
[b] Laboratory of Artificial Intelligence & Information Analysis, Department of Informatics, Aristotle University of Thessaloniki, 54 124 Thessaloniki, Greece
[c] Laboratory of Physiology, School of Medicine, Aristotle University of Thessaloniki, 54 124 Thessaloniki, Greece

ARTICLE INFO

Article history:
Received 30 November 2009
Received in revised form 9 March 2010
Accepted 20 April 2010

Keywords:
Spike sorting
ISOMAP
Extreme Learning Machine
Semi-supervised data learning

ABSTRACT

Background noise and spike overlap pose problems in contemporary spike-sorting strategies. We attempted to resolve both issues by introducing a hybrid scheme that combines the robust representation of spike waveforms to facilitate the reliable identification of contributing neurons with efficient data learning to enable the precise decomposition of coactivations.

The isometric feature mapping (ISOMAP) technique reveals the intrinsic data structure, helps with recognising the involved neurons and, simultaneously, identifies the overlaps. Exemplar activation patterns are first estimated for all detected neurons and consecutively used to build a synthetic database in which spike overlaps are systematically varied and realistic noise is added. An Extreme Learning Machine (ELM) is then trained with the ISOMAP representation of this database and learns to associate the synthesised waveforms with the corresponding source neurons. The trained ELM is finally applied to the actual overlaps from the experimental data and this completes the entire spike-sorting process.

Our approach is better characterised as semi-supervised, noise-assisted strategy of an empirical nature. The user's engagement is restricted at recognising the number of active neurons from low-dimensional point-diagrams and at deciding about the complexity of overlaps. Efficiency is inherited from the incorporation of well-established algorithms. Moreover, robustness is guaranteed by adaptation to the actual noise properties of a given data set. The validity of our work has been verified via extensive experimentation, using realistically simulated data, under different levels of noise.

1. Introduction

Spike-sorting procedures are based on the assumption that all the action potential traces of a particular neuron have nearly the same amplitude and shape. In extracellular recordings, the shapes of recorded spike waveforms mainly depend on a neuron's geometry and on its distance to the recording electrode. The goal of a spike-sorting routine is to process and analyse the usually composite recorded signals to identify the number of active neurons and extract detailed time courses of their spiking activity. Related algorithms constitute the core methodological component in various situations, ranging from traditional neurophysiological experiments and clinical/neuroscience studies to cortex–machine interfaces.

The pursuit of a sufficient spike-sorting technique is still an open issue, as indicated by the number of laboratories where the manual way of spike counting is still considered a competitive alternative. There is an inherent difficulty in the spike-sorting problem, which derives from the nature of the experimental data and delays further development. That is, the ground-truth regarding the origin of neural activity is simply unknown to the researcher. As a workaround, experimentalists have introduced the use of joint intra-extracellular recordings (Wehr et al., 1999; Harris et al., 2000). However, this approach does not constitute a panacea because it does not apply to the general case in which multiple neurons contribute to the signal.

The battery of available spike-sorting routines includes mainly automated techniques that analyse the recorded signals by means of their waveforms. At the initial stage, various linear techniques such as principal component analysis (PCA) and wavelets are included to reduce the dimensionality of the input data and enhance the signal content in the attempted representation. PCA-based projection is often restrained within the subspace spanned by the first two or three principal components (Glaser and Marks,

* Corresponding author. Tel.: +30 2310 998261/991839; fax: +30 2310 998269.
E-mail addresses: dadam@bio.auth.gr, d.adamos@ieee.org (D.A. Adamos), laskaris@aiia.csd.auth.gr (N.A. Laskaris), kosmidef@med.auth.gr (E.K. Kosmidis), theophil@bio.auth.gr (G. Theophilidis).

0165-0270/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.jneumeth.2010.04.018

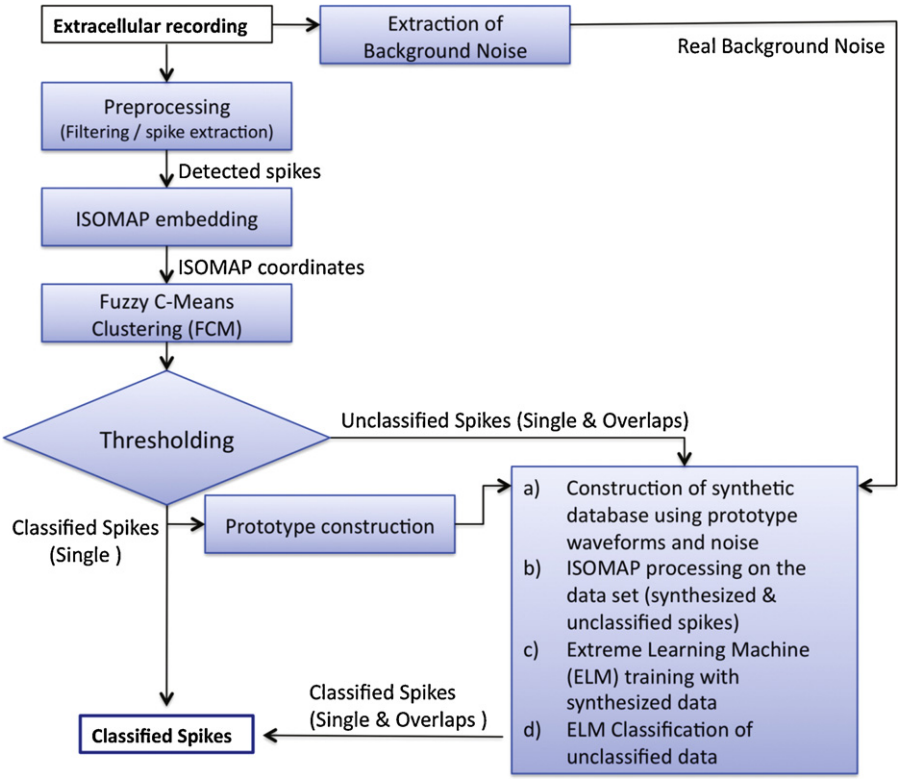

Fig. 1. NASS flow-chart.

1968; Abeles and Goldstein, 1977), although the employment of more components has recently been reported to carry useful complementary information (Adamos et al., 2008). Alternatively, the wavelet transform is used for the decomposition of spike waveforms (Letelier and Weber, 2000; Hulata et al., 2002; Quian Quiroga et al., 2004), featuring improved discrimination of localised shape divergences. In both the above approaches, the representation scheme serves as a pre-processing for a clustering framework that would take over the detection of distinct signal sources (i.e., active neurons) and the isolation of the corresponding spiking contributions.

The presence of background noise and the existence of overlapping spikes are inescapable, in almost every recording of neural activity. Apparently, both factors influence the performance of spike sorting. The former includes instrumental noise and activity from distant neurons, while the latter refers to cofiring of active, nearby, neurons. Noise in neural recordings has been assigned a stationary Gaussian profile in previous studies (Lewicki, 1998; Harris et al., 2000; Pouzat et al., 2002; Zhang et al., 2004). However, it has been argued that, in the general form, background noise is non-stationary and non-Gaussian (Fee et al., 1996a). Actually, a complex correlated noise profile (with higher power in low frequencies due to synaptic coupling among neurons, superimposed field potentials and bursting neurons) would seem more plausible. Spike-sorting methods featuring hierarchical clustering (Fee et al., 1996b) and nearest neighbours interactions (Quian Quiroga et al., 2004) have been designed to work without posing any modelling constraints regarding noise.

Apart from noise, the number of involved neurons and the complexity of overlaps (i.e., the number of cofiring neurons) cannot be defined *a priori*. They depend, mostly, on the particular experimental setting (type of electrode, depth, etc.) and the targeted brain area. The failure in the identification and treatment of overlapping spikes has already been spotted as a confusing factor and as the leading cause for estimating false-positive correlations in the available neural data (Bar-Gad et al., 2001). Conventional methods do not treat overlaps. They simply mark these waveforms as segments not corresponding to any of the activated neurons and classify them as noise. In the pursuit of overlap resolution, template matching appears as the most popular technique (Zhang et al., 2004; Vargas-Irwin and Donoghue, 2007; Franke et al., 2009), although it suffers from principle limitations whenever the template waveforms are highly correlated (Franke et al., 2009). Beyond template matching, only two studies have tackled the issue of overlaps, based on spectral analysis (Wang et al., 2006) and hidden Markov models (Herbst et al., 2008).

In this study, we propose a generalised approach to resolve reliably the overlaps and simultaneously achieve high robustness to noise. Our point of departure was previously published work on the related problem of single-trial analysis (Laskaris and Ioannides, 2002; Laskaris et al., 2004, 2008; Zigkolis and Laskaris, 2009), in which evoked brain responses were detected organised, and visualised by exploiting the recent advances in non-linear dimensionality reduction. Specifically, we employ isometric feature mapping (ISOMAP), which is known to reveal the intrinsic data variation and is therefore expected to be insensitive to random variations due to noise. The resulting low-dimensional parameterisation of the waveform variation can facilitate the recognition of active neurons through the true degree of freedom in the data. Moreover, their cofiring appears as blending among the basic (prototypical) waveforms, which are signatures of the individual neurons. Hence, the detection of exemplars for all active neurons and the recognition of participating neurons in the formation of complex signals can be performed efficiently within an ISOMAP representation space. Our approach employs fuzzy *C*-means (FCM) for the former, and a neural network, namely the Extreme Learning

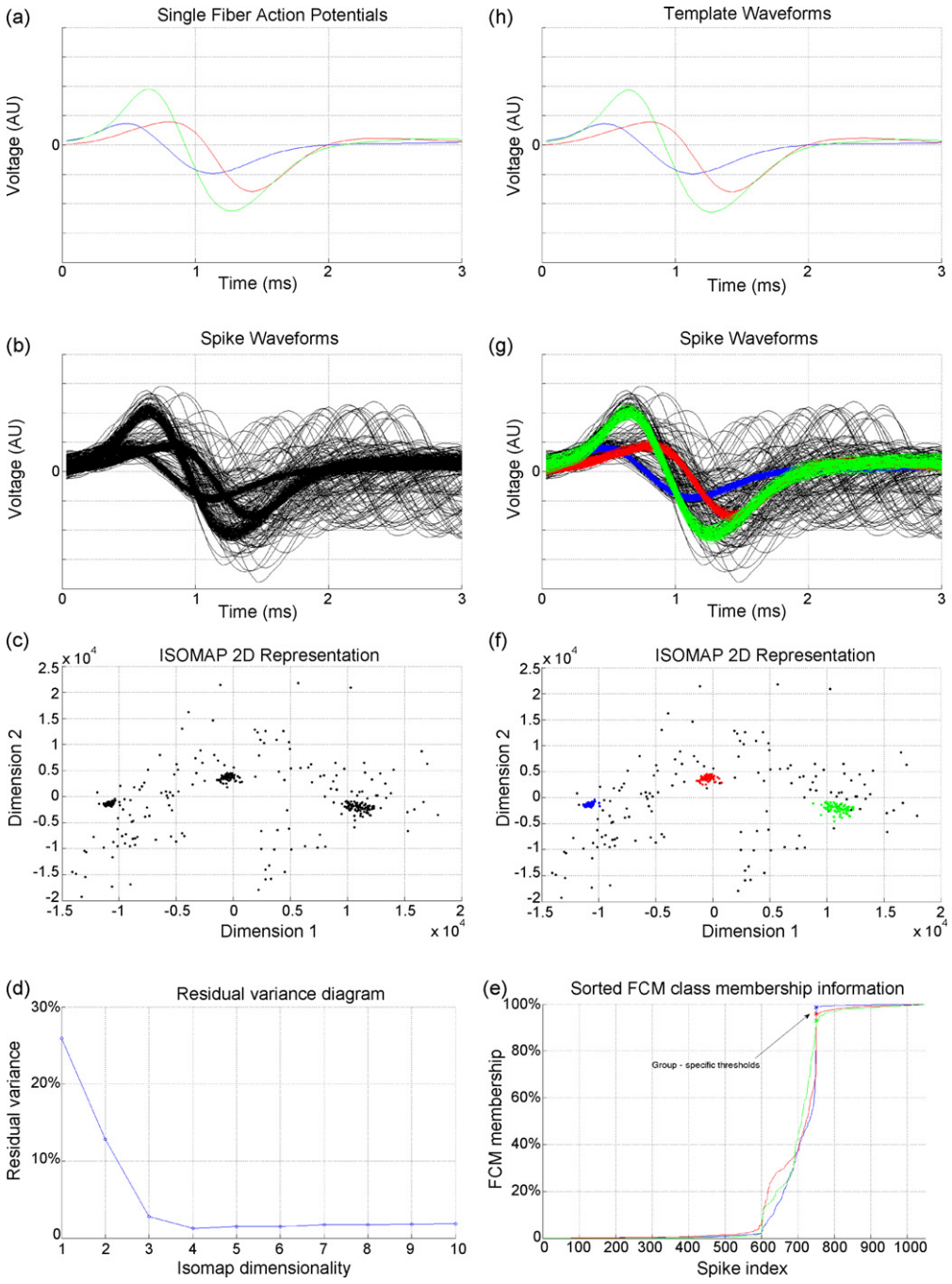

Fig. 2. The processing steps of main branch. (a) The waveforms used as templates to build the simulated data. (b) The overall simulated data set. (c) 2D ISOMAP representation of the data set. (d) Residual variance diagram of ISOMAP. The optimal dimensionality ($r_0 = 3$) is selected using the *elbow rule*. (e) Ranked FCM membership values for the three clusters. (f) Classification results visualized within 2D ISOMAP space. Colours indicate classes, while unclassified data are left black. (g) Classification results presented in the original data domain using the corresponding colours. (h) The derived prototypical spike waveforms.

Machine (ELM), for the latter. The user decides for the number C of active neurons and the complexity of overlaps, which is an integer ranging from 1 (denoting the absence of overlaps in the case of low firing rates) to 2 or 3 (denoting the number of potential contributors to a single waveform).

The novelty of our approach lies in the way the ELM is trained. Using prototypical waveforms, a synthetic database is constructed in which we simulate overlaps at the chosen level of complexity and add carefully extracted segments from the continuous recording, which are free of spiking activity. The network is first trained to decompose accurately these overlaps from their ISOMAP images and then applied to the actual overlaps. The generalisation ability of ELM, in conjunction with the robustness of ISOMAP, guarantees the successful spike identification and the high tolerance to background noise. Following a fully data-dependent strategy, we aimed at tailoring our hybrid spike-sorting system to the features of the given recording.

The remainder of this article is organised as follows. Section 2 introduces the new spike-sorting scheme, hereafter called NASS (Noise-Assisted Spike Sorting). Section 3 includes the evaluation of our method based on extensive experimentation with realistically simulated data. Section 4 concludes this article with a short discussion on some technical aspects and contains some ideas on possible extensions of this work. For the sake of completeness, the

individual algorithms composing the NASS-scheme are reviewed in the appendices.

2. Methodology

The architecture of the NASS framework includes two branches for the processing of (single and overlapping) spikes, as can be seen in the flow-chart of Fig. 1. The main branch deals with the whole bunch of the data, but addresses the classification of non-overlapping spike waveforms only. The remaining data, which are characterised by high degree of uncertainty and include noise-corrupted or overlapping waveforms, are handled at a subsequent stage following a delicate decision-making process.

In a nutshell, spike waveforms derived from the pre-processing step (dealing with the extraction of spikes from the continuous recordings) are mapped into a reduced (ISOMAP-based) low-dimensional coordinate space. Using this representation, the user identifies the number C of clearly formed clusters that can be associated with distinct active neurons. The subsequent application of FCM (within this coordinate space) associates with each waveform a set of membership values, measuring its affinity to the formed groups. Any spike carrying high membership values to one of the clusters, is marked as classified and will participate in the succeeding definition of the template waveform for the corresponding group. On the other hand, spike waveforms with low memberships (i.e., lower than a data-driven threshold) remain unclassified and will be directed to the second stage of NASS framework.

In the side processing, the derived exemplars from the previous stage are first used to build a synthetic database with known composition for each waveform. The included waveforms correspond to either single neuron activity or represent complex coactivation patterns (derived by merging two or more prototype waveforms with time lags systematically varied to simulate the spike overlap effect). These waveforms are masked by real noise extracted from the experimental data and then gathered together with the ambiguous unclassified spikes from the previous stage. Next, an ISOMAP representation is derived for the overall data set (both the synthetic data and the real unclassified spike waveforms). Living within this feature space, an ELM is then trained to perform the decomposition of the synthetic waveforms and finally decides on the particular decomposition of the real (unclassified) ones.

2.1. Main branch (in depth): reducing dimensionality, identifying neurons and data-sieving

The segments, extracted from the time series of extracellular recording via a root-mean-square threshold detector, are considered as the first raw representation of spiking activity. The ensemble of these loosely aligned spike waveforms can be thought as a point-swarm residing in a multidimensional feature space with axes corresponding to signal amplitudes at particular latencies. Following the standard convention, the ith spike waveform is depicted as $x_i(t)$, $t = 1, 2, \ldots, T$, $i = 1, 2 \ldots, N$ (with t denoting discrete time or latency) and represented via the row-vector $\mathbf{x_i} = [x_i(1), x_i(2) \ldots, x_i(t) \ldots, x_i(T)] \in R^T$. Similarly, the whole ensemble is represented in a data-matrix format as $\mathbf{X_{[N\times T]}} = [\mathbf{x_1}|\mathbf{x_2}| \ldots \mathbf{x_i}| \ldots \mathbf{x_N}]$.

Most often, an enhanced representation is sought by employing a linear transformation based on PCA to reduce the dimensionality of feature space and simultaneously to suppress random variation in the data. Besides its popularity, this tactic has some limitations. PCA assumes the orthogonality of the sources contributing to the variation of the waveforms, a condition that is rarely satisfied in practice. Its linear character might not be beneficial for the task of identifying distinctly active neurons. This is the main reason that we resort to a non-linear scheme for dimensionality reduction instead. In particular, we employ ISOMAP embedding (Tenenbaum et al., 2000) to achieve a parsimonious representation, in which the true degrees of freedom can be easily recognised and directly associated with involved neurons.

The algorithmic details of ISOMAP technique can be found in Appendix A. It starts by building a neighbourhood graph over the data points in the original feature space. This graph is then used to compute all the geodesic inter-point distances. Multidimensional scaling is finally employed to derive a reduced coordinate space where these distances are preserved and therefore the intrinsic geometry of the data is faithfully represented. In our case, the ISOMAP-routine provides a geometrical picture, within an r–D space, of the spike waveforms variation:

$$\mathbf{Y_{[N\times r]}} = [\mathbf{y}_1|\mathbf{y}_2| \ldots \mathbf{y}_i| \ldots \mathbf{y}_N] = \mathrm{ISOMAP}(\mathbf{X}, r),$$
$$\text{where } \mathbf{y}_i = [y_i(1), y_i(2), \ldots, y_i(r)] \in R^r \quad (1)$$

The derived point-swarm is accompanied with the 'residual variance', which is a performance index (Samko et al., 2006) ranging from 0% to 100% and indicating the reliability of the mapping (for an example, see Fig. 2(d)). The optimal dimensionality r_0 can be sought (as a compromise between accuracy and compression) by computing multiple maps with increasing r ($r \in [1, 10]$), drawing the diagram of 'residual variance' as a function of r and applying the 'elbow rule'.

ISOMAP is implemented here in its 'k-nearest neighbours' mode using an advanced version of the algorithm (i.e., version II), which takes advantage of sparseness in the emerging graph and redundancy in the distances (Tenenbaum et al., 2000). In the general case, we adopt a high 'k' value, around 1/5 of the total number of spike waveforms. In cases of very low signal-to-noise ratio (SNR) recordings, however, a lower value for k (e.g., 5–10) is suggested to enhance the clustering tendencies in the resulting map. Using the two-dimensional (2D) (or three-dimensional (3D)) ISOMAP representation, the user identifies the number of active neurons C as the number of clearly formed clusters in the whole point-swarm.

On the other hand, the r_0–D ISOMAP representation is used for the execution of FCM algorithm with C centres, to group the spike waveforms into classes of similar morphology (and therefore common origin). FCM operation is denoted compactly as

$$\mathbf{U_{[N\times C]}} = \mathrm{FCM}(\mathbf{Y}, C) \quad (2)$$

FCM associates with the jth spike-waveform a set of membership values ($u_{j1}, u_{j2}, \ldots, u_{jc}, \ldots, u_{jC}$) measuring its similarity with the formed groups. Each waveform can be assigned to one of the classes, based on the maximum membership defuzzification step (Zouridakis and Tam, 2000). This rule is, however, hardened here so as to ensure that only waveforms of simple composition will participate in the subsequent derivation of signatures for the involved neurons. For this purpose, the membership values for each group are sorted and a group-specific threshold $thr_1(c)$, $c = 1, 2, \ldots, C$, is estimated as the value corresponding to the leftmost point of the plateau at a height of 100% (see Fig. 2(e)). Before the assignment of waveforms to groups, the maximum membership value $u_{j\max} = \max(u_{j1}, u_{j2}, \ldots, u_{jc}, \ldots, u_{jC})$ of the jth waveform is compared against the threshold corresponding to the group with the highest affiliation. If $u_{j\max}$ does not exceed the corresponding group-specific threshold and a second common threshold $thr_2 = 0.85$, the waveform is directed to the second stage of analysis. The rest of the waveforms are used to derive (by weighted averaging) prototypical waveforms for each of the C neurons. The motivation of using the group-specific threshold is to stay within the core of each group of spike waveforms, while the second threshold is activated in the case of low-SNR scenarios. In a strict

mathematical notation, this data-sieving step is denoted as follows:

$$m_j^c = \left\{ \begin{array}{ll} u_{j\max} & \text{if } u_{j\max} > thr_1(c) \text{ and } u_{j\max} > thr_2 \\ 0 & \text{otherwise} \end{array} \right\} \quad (3)$$

$$\mu_j^c = \left\{ \begin{array}{ll} 1 & m_j^c > 0 \\ 0 & \text{otherwise} \end{array} \right\} \quad (4)$$

The first variable m_j^c is the modified membership of jth waveform to the cth cluster of highest affiliation, while the second is an indicator function denoting the acceptance of the waveform as classified (to any of the formed groups).

If we denote the population for each one of the well-trimmed groups (containing only clearly classified spike waveforms) as N_c, $c = 1, 2, \ldots, C$, and the population for unclassified spikes as N_0, the following holds:

$$N_c = \sum_j \mu_j^c, \qquad \sum_{c=1}^{C} N_c + N_0 = N \quad (5)$$

The corresponding prototypical spike waveforms are computed as

$$\text{prototype}_c = \frac{\sum_{j=1}^{N} (m_j^c \cdot \mathbf{x}_j)}{\sum_j m_j^c}, \quad c = 1, 2, \ldots, C \quad (6)$$

The above ideas are demonstrated with the example included in Fig. 2 and is based on artificial data constructed to simulate data from an extracellular neural recording. These data were generated using the three actual single fibre action potential signals (SFAPs) shown in Fig. 2(a). The SFAPs were extracted from experimental extracellular recordings from the peripheral nervous system of the beetle *Tenebrio molitor*. By replicating 300 times each SFAP (i.e., a set of 3×300 waveforms) and adding 50 randomly overlapping spikes for every possible combination between the three SFAPs (i.e., a set of 3×50 waveforms), we formed the complete data set of 1050 waveforms, which is presented in Fig. 2(b) (together with a portion of real background noise, extracted from the recording). The 2D ISOMAP representation of this dataset is shown in Fig. 2(c). Considering the high density areas of this point-diagram, there are apparently three clusters (pointing at the existence of three active neurons). The residual variance graph, which is a supplementary output of the ISOMAP routine, is included in Fig. 2(d) and clearly designates the first three dimensions as necessary for the faithful low-dimensional representation of the data. Hence, the first three ISOMAP coordinates were used to represent each spike waveform (for convenience, only the first two are shown) and FCM was applied to the new data-matrix $\mathbf{Y}_{[1050\times4]}$ with $C = 3$ centres. For the defuzzication of the resulting membership matrix $\mathbf{U}_{[1050\times3]}$, spike memberships were sorted separately for each of the three classes, as shown in Fig. 2(e), and group-specific thresholds were derived. Spikes with highest membership exceeding the previous thresholds (and higher than the common $thr_2 = 0.85$ threshold) were assigned to the corresponding classes, while the rest were left unclassified. This data-sieving procedure step is visualised in ISOMAP space (Fig. 2(f)) and in the original data domain (Fig. 2(g)) using a three-level colour code. The corresponding prototypical spike waveforms are shown in Fig. 2(h), making evident their resemblance with the original SFAP waveforms (see Fig. 2(a)).

2.2. Side branch (in depth): employing an ELM to classify the ambiguous spikes

Ambiguous spike waveforms, failing to pass the previous data-sieving step, are directed to a single-layer forward network trained via the ELM algorithm (see Appendix B). The choice of this novel classification scheme was motivated by its computational elegancy and fast-learning capabilities, which lead to competitive performance with respect to other contemporary learning algorithms like back propagation neural networks (BPNNs), radial basis function networks (RBFNs) and support vector machines (SVMs), as it has been reported for similar classification problems (Kim et al., 2009). The well-known advantage of high generalisation performance constitutes the ELM suitable for, first, training with an appropriate set of prototypical spike waveforms and then applying to previously unseen spike waveforms to extend robustly the decision function underlying in the main branch.

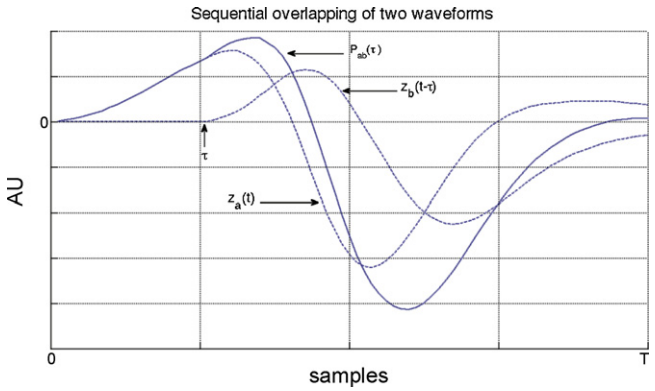

Fig. 3. An overlap pattern (P_{ab}) emerging from the superposition of two waveforms (z_a and z_b) with a time-lag τ.

In this work, ELM is set to work within ISOMAP representation space and acquires knowledge in a very principled manner by selecting carefully the training data set. The main idea is to build a synthetic database from the prototypical spike waveforms defined by the algorithmic operations in the main branch. The true labels of the synthesised waveforms are therefore known (and in the case of single/overlapping spikes, indicate their classification/composition) and ELM can learn the association between their ISOMAP images and their composition. By applying the trained ELM to the real waveforms left unclassified at the first stage of processing, a delicate full characterisation of all spike waveforms can be accomplished. Every single waveform in the synthetic data set is built by blending two (or more) prototypes. This procedure is described in mathematical terms by using each of the estimated prototypical waveforms to define finite signals:

$$z_c(t) = \left\{ \begin{array}{ll} \text{prototype}_c(t) & t \in [0, T] \\ 0 & \text{otherwise} \end{array} \right\} \quad (7)$$

which are then combined to derive a waveform that simulates the cofiring of two neurons:

$$P_{ab}(\tau) = [z_a(t) + z_b(t - \tau)]|_{t=0}^{t=T}, \quad 0 \le \tau \le T \quad (8)$$

By changing the time-parameter τ, different overlaps can appear (resembling to spiking activity waveforms that remained unclassified during the execution of the main branch). A characteristic example is shown in Fig. 3. Eq. (8) describes a waveform, which is defined in a discrete time window $[0, T]$ and results from the superimposition of signal $z_b(t)$ to signal $z_a(t)$ with a phase difference (time lag) of τ samples.

A synthesised database containing overlaps from two neurons (i.e., corresponding to the lowest complexity scenario) is constructed by forming all possible combinations: $P_{12}(\tau)$, $P_{21}(\tau)$, $P_{13}(\tau)$, $P_{31}(\tau)$, …, $P_{1C}(\tau)$, $P_{C1}(\tau)$, …, $P_{23}(\tau)$, $P_{32}(\tau)$, …, $P_{2C}(\tau)$, $P_{C2}(\tau)$, …, $P_{(C-1)C}(\tau)$, $P_{C(C-1)}(\tau)$, where $\tau = 0, 1, \ldots, T$. According to Eq. (8), the following holds:

$$P_{ab}(T) = [z_a(t) + z_b(t - T)]|_{t=0}^{t=T} = \{z_a(1), z_a(2), \ldots, z_a(t), \ldots, [z_a(T) + z_b(0)]\} \approx z_a(t), \quad \text{when } z_b(0) \approx 0 \quad (9)$$

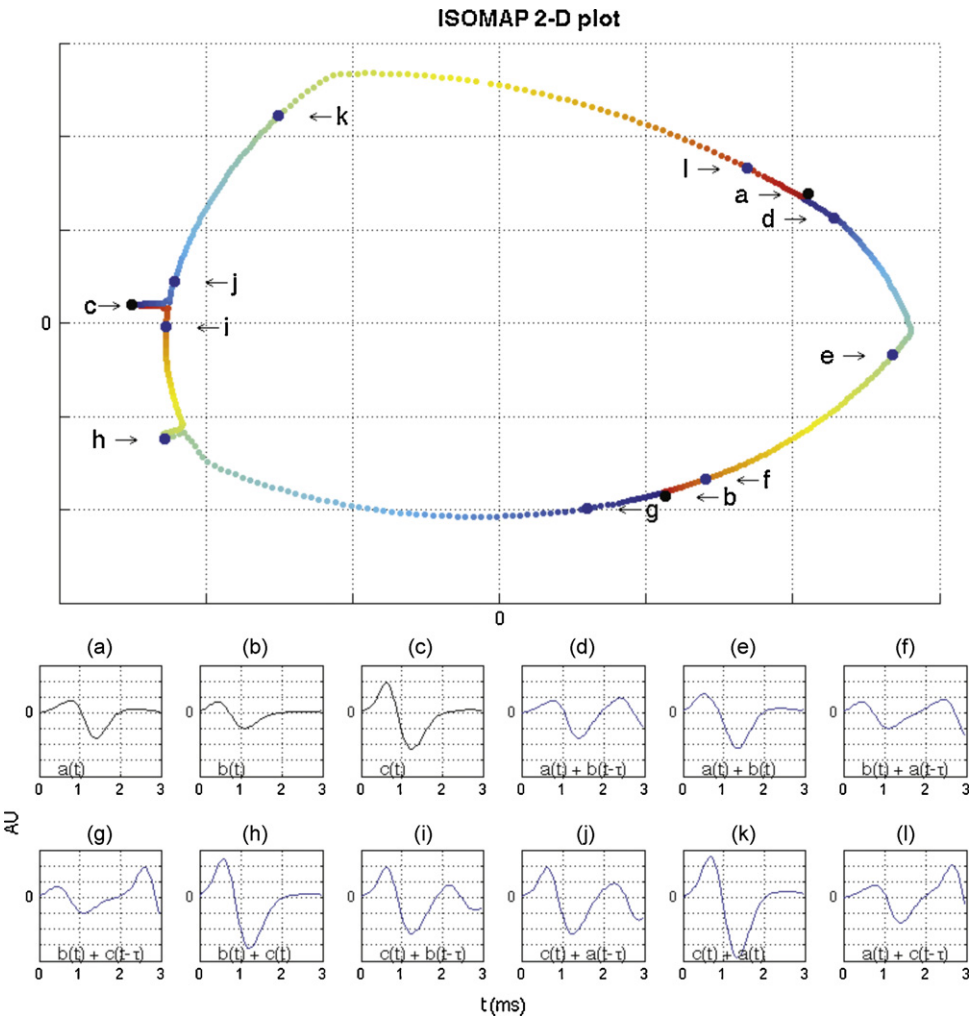

Fig. 4. In the upper panel, the 2D ISOMAP representation is provided for an exemplar set of noise-free synthetic overlaps (z_i), which were created using the three prototypes shown in Fig. 2(h). The waveforms corresponding to selected points (a–l) of this closed curve are provided below.

In words, this means that the addition of $z_b(t)$ with the maximum time-lag leaves the $z_\alpha(t)$ signal intact (e.g., waveforms *a*, *b* and *c* in Fig. 4). Hence, the synthetic database includes prototypical spiking activity from single neurons as limiting cases of the overlapping simulation procedure.

The above ideas for building a synthetic database can easily be generalised in the case of triple overlaps. For instance, overlaps from three neurons can be defined using two time-lag parameters:

$$P_{abc}(\tau_1, \tau_2) = [z_a(t) + z_b(t-\tau_1) + z_c(t-\tau_2)]|_{t=0}^{t=T}, \quad 0 \le \tau_1, \tau_2 \le T \tag{10}$$

while,

$$P_{abc}(T, T) = [z_a(t) + z_b(t-T) + z_c(t-T)]|_{t=0}^{t=T} \approx z_a(t), \quad \text{when } z_b(0),\ z_c(0) \approx 0 \tag{11}$$

The final composition of this database can be tailored to the given experimental setting so as to avoid unnecessary computations (as described in Section 3). In this article, we adopted a very simple rule. An ensemble of $(C^2 - C) \times T$ waveforms was first derived using Eq. (8). This set was augmented by $[(C^2 - C)/10] \times T$ waveforms produced via randomly chosen values of τ_1, τ_2 in Eq. (10). Higher complexity overlaps were considered very rare and were not treated here. In every case, the synthetic waveforms are accompanied with classification labels indicating the (unique combination of) contributing neurons.

Using the ISOMAP representation of the previous database, ELM will be trained to perform the identification of noisy-spikes origin and the overlap decomposition. A typical example of such a representation is shown in Fig. 4. In the particular example, only overlaps from pairs of neurons were realised. The apparent parameterisation corresponds to the actual composition of the synthetic waveforms and this association is a trivial task for almost any classifier. However, since our goal is to devise a robust procedure that can perform sufficiently when dealing with realistic experimental data (i.e., low SNR), we use, during the ELM training, noisy versions of the synthetic waveforms as well. Apart from adding robustness, this will facilitate the selection of the only free-parameter of the ELM (that is, the number of neurons L in the hidden layer (see Appendix B)) based on the generalisation performance.

Therefore, we replicate the synthetic waveforms and add extracts of real background noise from our experimental data. The waveforms in this additional data set carry the classification labels of their predecessors and will be used to measure the performance of ELM after training with the first idealised (i.e., noise-free) synthetic database. The waveforms from the two synthetic databases,

together with the real unclassified spike waveforms, are brought in the same ISOMAP space. By denoting with $\mathbf{z}_i$, $\mathbf{z}_i^*$ and $\mathbf{x}_j$, respectively, the idealised synthetic overlaps, the noisy synthetic overlaps and the real unclassified waveforms, the mapping procedure can be written as

$$\mathbf{Y}' = \mathrm{ISOMAP}(\mathbf{X}', r_0), \tag{12}$$

where

$$\begin{aligned} X' &= [\boldsymbol{z}_1|\boldsymbol{z}_2|\dots\boldsymbol{z}_{N_z}|\boldsymbol{z}_1^*|\boldsymbol{z}_2^*|\dots\boldsymbol{z}_{N_z}^*|\boldsymbol{x}_1|,\boldsymbol{x}_2|\dots\boldsymbol{x}_{N_0}] \text{ and } \mathbf{y}'_{[(2N_z+N_0)\times r]} \\ &= [\mathbf{y}'_1|\mathbf{y}'_2|\dots y'_{2N_z+N_0}] \\ &= [\boldsymbol{y}_1^z|\boldsymbol{y}_2^z\dots\boldsymbol{y}_{N_z}^z|\boldsymbol{y}_{N_z+1}^{z*}|\boldsymbol{y}_{N_z+2}^{z*}|\dots\boldsymbol{y}_{2N_z}^{z*}|\boldsymbol{y}_{2N_z+1}^{x}|\boldsymbol{y}_{2N_z+2}^{x}|\dots\boldsymbol{y}_{2N_z+N_0}^{x}] \end{aligned}$$

For the automated crafting of ELM (i.e., definition of parameter L), we employ a composite training scheme that proceeds in an iterative manner with increasing the number of neurons in the hidden layer. For each L, the training is repeated 10 times (each time with a different initialisation for the weights and biases of the hidden layer). We have incorporated a cross-validation tactic within each primitive training step, to ensure high generalisation performance. At each iteration, the $\mathbf{z}_i$ vectors are used to estimate the network parameters (the matrix $\mathbf{B}$ (see Appendix B)), and then the $\mathbf{z}_i^*$ vectors are fed to the derived network to measure its classification performance. The procedure terminates by reporting the particular network (with all its parameters fixed) that achieved the best classification of the $\mathbf{z}_i^*$ waveforms. This network is the final classifier that will act on the unclassified real spike waveforms $\mathbf{x}_j$ to complete the spike sorting. It has been trained so as to generalise, in the most robust way, the decision-making process of the first branch.

In an algorithmic fashion, the overall ELM training proceeds as follows:

Algorithm. ELM Training

1: for $L = 1$ to $q_N \cdot N$: $q_N \in (0, 1)$

2: for $k = 1$–10:

3: TRAIN (k): Train ELM (L) with [ISOMAP coordinates|labels of origin] of noise-free synthetic waveforms ($[\mathbf{y}'_1|\mathbf{y}'_2|\dots\mathbf{y}'_{N_z}]$)

4: TEST (k): Classify [ISOMAP coordinates] of the noisy synthetic waveforms ($[\mathbf{y}'_{N_z+1}|\mathbf{y}'_{N_z+2}|\dots\mathbf{y}'_{2\times N_z}]$) with the trained ELM ($L$) and estimate success rate comparing [ELM classification labels | labels of origin]

5: For the highest TestingAccuracy (k), save the trained ELM(L) (L, $\mathbf{H}$ ($\mathbf{w}_i$, b_i) and β_i)

6: end

7: For the highest TestingAccuracy (L), save the trained ELM(L) (L, $\mathbf{H}$ ($\mathbf{w}_i$, b_i) and β_i)

8: end

The ELM training procedure is demonstrated, in a very simple case (where cofiring of more than two neurons can be ignored and no single spikes has been left unclassified during the first stage of processing), in Figs. 4 and 5. Fig. 4 includes the 2D ISOMAP representation of the noise-free synthetic waveforms that were created based on the three prototypes originally shown in Fig. 2(h). Points *a*–*c* represent prototypical spike waveforms, which are shown in adjacent Fig. 4(a)–(c). Points *d*–*f* represent the progressive blending between waveforms *a*–*b*, while waveform *e* corresponds to zero-phase superimposition. Similarly, the blending between the firing of remaining neurons appears as paths connecting the *b*–*c* and *c*–*a* points.

Since, in this particular case, the unclassified spike waveforms do not include overlaps of complexity 3, an ELM-network with six

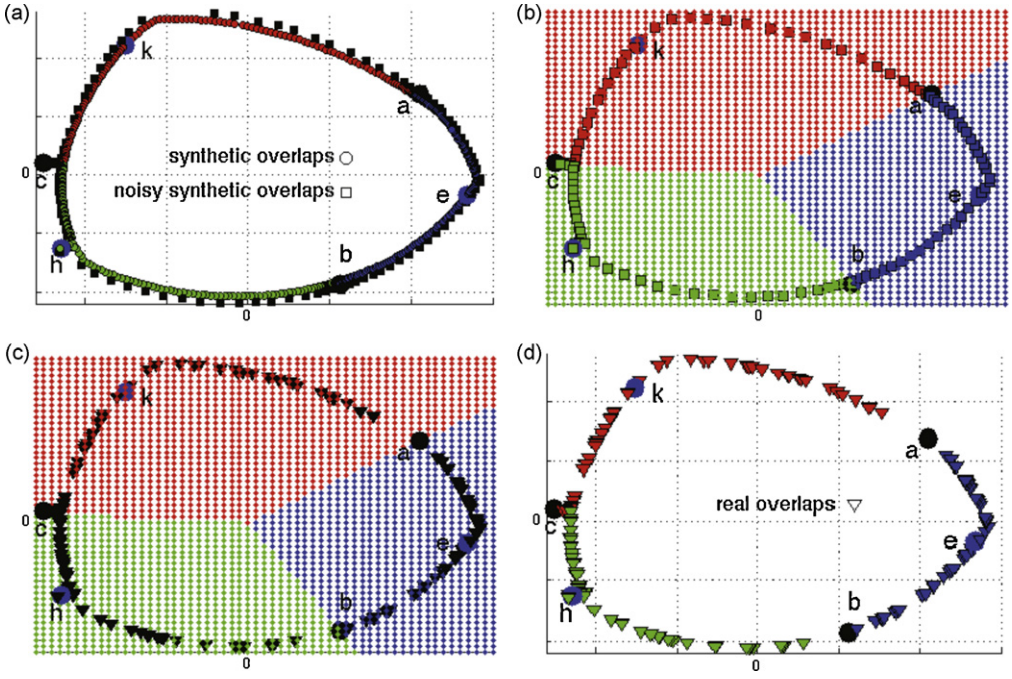

Fig. 5. The processing steps of the side branch are demonstrated using the unclassified spikes (black waveforms) from Fig. 2(g). (a) ISOMAP images of noise-free synthetic overlaps z_i (○) (coloured according to their classification labels) together with a portion of noisy synthetic overlaps z_i*(□). ELM will be trained until all z_i^* are correctly classified. (b) ELM training is complete; all the z_i^* (□) waveforms are assigned their correct classification label. The decision function surface for the optimal ELM-network is drawn in the ISOMAP coordinate space. (c) Images of unclassified waveforms x_i (▽) are shown together with the ELM decision function surface. (d) ELM classification has been completed; images of unclassified waveforms x_i (▽) are now carrying classification labels.

output nodes will suffice. Each output neuron will act as a detector of a spike from a particular neuron {1, 2, 3} or a particular overlap {1–2, 2–3, 3–1}. Fig. 5(a) depicts the ISOMAP images of synthetic overlaps $\mathbf{z}_i$, coloured according to the corresponding classification label. The corresponding images of single prototypical spikes are also included, as shown in equation 10, and illustrated by the black point-areas denoted by '*a*, *b*, *c*'. These points will be used during the training step. In addition, a portion of noisy synthetic overlaps $\mathbf{z}_i^*$ is also included in black (since these waveforms will be used during the test step). The instantiation included in Fig. 5(b), corresponds to the test step during which the $\mathbf{z}_i^*$ waveforms are assigned a classification label. The decision function surface for the optimal ELM-network is drawn in the ISOMAP coordinate space and the labelling of the test-set is indicated (corresponding to 100% success in the recognition rate). In Fig. 5(c), the images of unclassified waveforms $\mathbf{x}_i$ are shown in the ISOMAP coordinate space, together with decision function surfaces of the optimal ELM. The function of this network is visualised in Fig. 5(d).

3. Results

We first generated spike-waveform data sets with realistic features (described in Section 3.2) to evaluate the performance of the NASS-scheme under different SNRs (Section 3.3) and compare it against well-known techniques (PCA – Expectation–Maximisation (EM), wavelets-super paramagnetic clustering (SPC)). In addition, we borrowed realistic continuous-time data sets of various SNR (Quian Quiroga et al., 2004) to demonstrate the performance of NASS in publically available data sets (Section 3.4).

3.1. Overlap complexity

We estimated the complexity of overlaps resulting from a variable number (ranging from 2 to 6) of tonic firing neurons. Their inter-spike intervals followed a dead time Poisson distribution with a refractory period of 4 ms. We considered typical values for the firing rate corresponding to different brain areas and our laboratory data: 60 Hz for Globus Pallidus, 25 Hz for subthalamic nucleus (STN), 5 Hz for the cortex (Bar-Gad et al., 2001) and 20 Hz for our data. The results showed that the presence of triple overlaps is significant only in the case of 60 Hz, for more than three tonic firing neurons. Quadruple overlaps can be ignored in all cases.

3.2. Artificial waveform dataset

For the detailed evaluation of our method, we used datasets of artificial waveforms. In this way, we have avoided additional errors introduced by the pre-processing step of spike detection. This step is essential in the processing of continuous time series and is taken into consideration in the evaluations included in Section 3.4.

Aiming at realistic simulations, we generated noisy spike waveforms representing neural activity from three distinct neurons. Different data sets were constructed with fixed SNR, ranging from 1.5 to 6. We used real action potentials from respiratory motoneurons which had been recorded *in vitro* with a single 'hook' electrode from the peripheral nervous system of the beetle *Tenebrio molitor* (Zafeiridou and Theophilidis, 2004). Three such action potential waveforms served as the initial templates. These waveforms had been recorded extracellularly with a sampling frequency of 30 kHz and time duration of 4 ms (120 samples). The templates were replicated multiple times and added to segments of background noise extracted from the same recording (randomly extracted from latencies during which the spike-detector was silent). To pursue evaluation results under different SNR levels, each extract of real background noise was modulated by a variable, positive amplitude factor, φ. The SNR of the resulting waveform (template plus noise segment) was then defined as follows:

$$\mathrm{SNR} = \frac{\mathrm{RMS(of\ template\ waveform)}}{\varphi \cdot \mathrm{RMS(of\ random\ noise\ extract)}} \tag{13}$$

Three-hundred (300) waveforms per template were generated yielding 900 single spikes corresponding to the three neural classes of our data set (see Fig. 6(b)). In addition, paired combinations among the three templates were realised by first inducing variable delays and then adding noise segments using amplitude factors to achieve a given SNR level. In this way, 150 waveforms of double overlaps were generated; 50 for each template pair (see Fig. 6(c)–(e)). This dataset of 1050 waveforms is denoted as Dataset-1 with an indication for the SNR level. Finally, 50 more waveforms (seen in Fig. 6(f)) were added to the previous data set, corresponding to triple overlaps with their SNR level adjusted accordingly. The complete second data set of 1100 waveforms is also accompanied by an indication for the SNR level; it is denoted as Dataset-2 and is shown in Fig. 6(a).

3.3. Performance evaluation

Type I (false-positive (FP))/II (false-negative (FN)) errors (Harris et al., 2000) are widely used metrics in the related literature for spike sorting. They derive from the traditional classification schemes and conceptualise misclassification. Provided that the actual origins of the spike waveforms under study are known, a false-positive (FP) corresponds to a spike wrongly assigned to the set of waveforms for a particular neuron. Likewise, a false-negative (FN) corresponds to a spike missing from the set of a neuron's spike waveforms. However, in practice, the previous definitions have to be modified, since in real extracellular recordings the ground-truth is unknown. Hence, the notion of FP/FN is influenced by practical issues such as the existence of outliers or the existence of untreatable overlaps. Usually, unidentified waveforms are considered as an additional class and the two errors are estimated based on the rest. As a result, the FN error in spike-sorting routines without overlap resolution is underestimated.

We chose to comparatively evaluate the performance of NASS, for various SNR levels, employing two separate validation modes, denoted hereafter as 'class-based' and 'neuron-based' validation. The first mode reflects the conventional practice in the spike-sorting scene, which is based on the separation of the well-identified waveforms (most probably single spikes) from the rest (which are treated as noise). Following this strategy, the number of active neurons is to be initially identified; then, the well-identified waveforms are classified into separate classes (one for each neuron). The total number of FP and FN is finally referred to the identified 'classes' and used to define measures for the evaluation of spike-sorting routine. This validation tactic shares the above classification weakness, since overlaps are not treated on the basis of their contributing source waveforms but are simply characterised as noise. Thus, the initial waveforms participating in an overlap are neither appearing nor incorporated in the evaluation procedure.

Beyond this approach, we chose to additionally employ another validation scheme closer to the ground-truth: the 'neuron-based' validation mode. Here, spike waveforms are validated on the basis of their source neurons. Single-spike waveforms are assigned to the corresponding neurons, while overlaps are first resolved and then used to update the populations of the contributing neurons, accordingly. Thus, the total number of FP and FN is now referred to the identified neurons and defines more plausible measures for the evaluation of the spike-sorting routine.

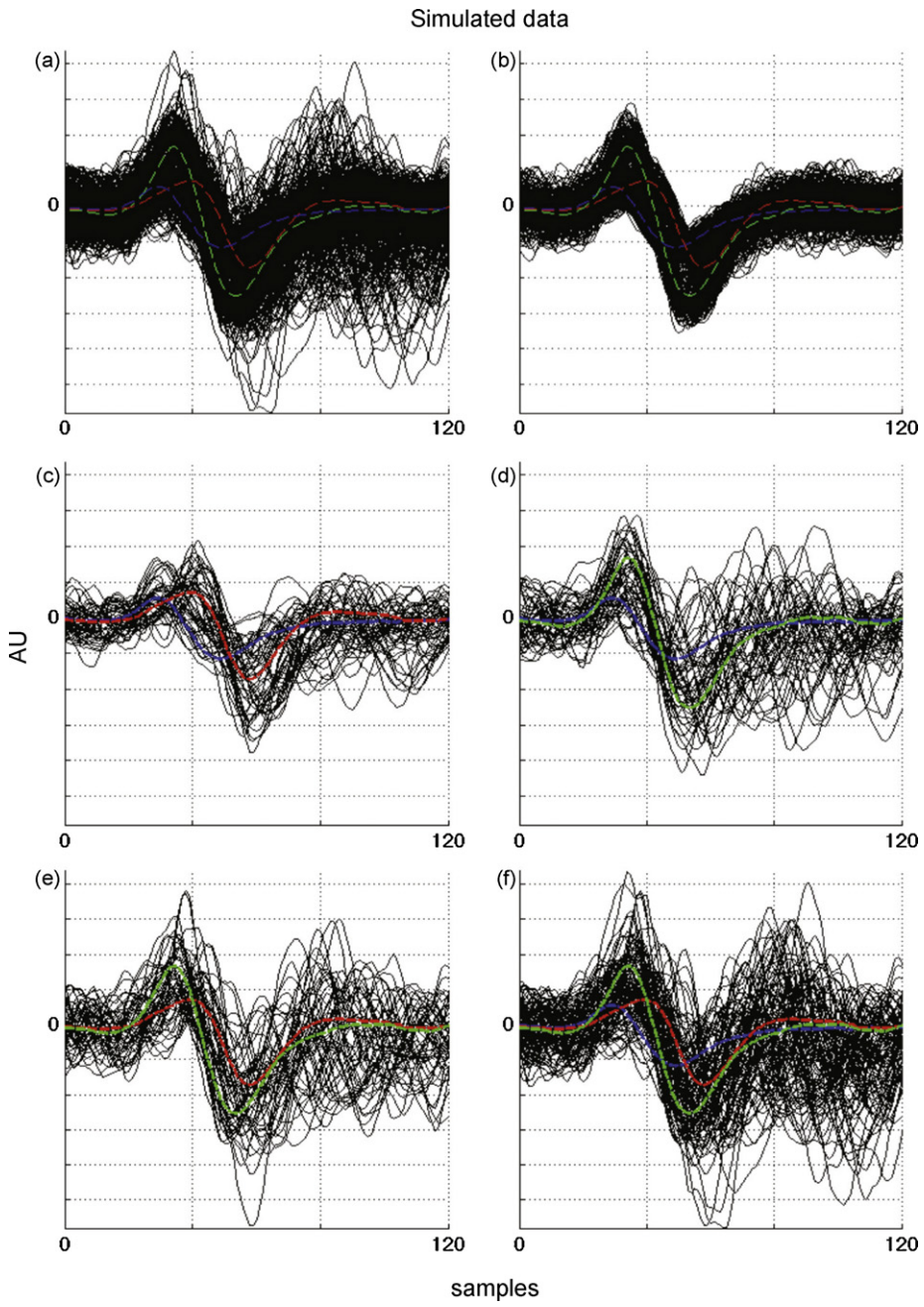

Fig. 6. Data set-1: (a) the complete data set of 1100 waveforms. (b–e) The included double overlaps (50 waveforms for each case). (f) The triple overlaps (50 waveforms). The SNR is 2 for all waveforms.

In either case, the adopted 'error rate' accounts for the total number of FP and FN spikes:

$$\text{error-rate} = \frac{\sum_i (Fp_i + Fn_i)}{\text{total \# of participating waveforms}} \cdot 100\% \qquad (14)$$

with i running over the number of single-spike classes (i.e., the number of identified neurons). It should be noticed that the denominator differs in the two modes. Tables 1 and 2 depict indicative instantiations of FP/FN measurements carried out according to both modes.

Before evaluating the overall NASS framework, we have included – as an intermediate evaluation step – the comparison of ISOMAP representation against the PCA representation. The obtained results are shown in Fig. 7(a). For both methods, the clustering step was performed using FCM (as it was presented in Section 2.1). In the case of PCA, four principal components (Adamos et al., 2008) had been used. For completeness, the same clustering step was applied to the raw waveforms as well. It is evident that both dimensionality reduction techniques contribute to the discrimination task, while ISOMAP is the favourable representation.

The performance of NASS framework was studied, in a comparative fashion, against two popular spike-sorting techniques, using both previously defined validation schemes. The first technique employs PCA for data representation and Expectation–Maximisation clustering (the EM classification was realised using 'Klustakwik' (Harris et al., 2000) and implemented using 'the enhanced version of nev2lkit' (http://neurobot.bio.auth.gr/nev2lkit – Adamos et al., 2008)). The second technique uses wavelet representation and SPC (it was realised using 'Waveclus'

Table 1
Class-based validation mode: an example of results derived for **Dataset-2**.

Simulated data	Spikes			
	1100			
Templates	Classes			
	Class I	Class II	Class III	Noise
Template I	300			
Template II		300		
Template III			300	
Template I + II				50
Template I + III				50
Template II + III				50
Template I + II + III				50

Waveclus	Spikes	FP	FN	Error rate
	1100	33	0	3.00%

Templates	Classes			
	Class I	Class II	Class III	Noise
Template I	300			
Template II		300		
Template III			300	
Template I + II	13	9		28
Template I + III			6	44
Template II + III		2	2	46
Template I + II + III			1	49

NASS	Spikes	FP	FN	Error rate
	1100	1	0	0.09%

Templates	Classes						
	Class I	Class II	Class III	Class I + II	Class I + III	Class II + III	Class I + II + III
Template I	300						
Template II		300					
Template III			300				
Template I + II				50			
Template I + III					42	1	7
Template II + III						41	9
Template I + II + III			1	2	5	7	35

(http://www.le.ac.uk/neuroengineering/software – Quian Quiroga et al., 2004)).

Adopting the class-based validation mode and using the two datasets (Dataset-1 and Dataset-2), both conventional methods classified the spikes into three main classes and an additional noise-class (for an example, see Table 1 – Waveclus). FPs and FNs were categorised on the basis of three defined classes. On the other hand, the NASS-scheme proceeded with the overlaps one step further. It defined separate classes for each overlap (double and triple) as can be seen in Table 1 – NASS section. To ease comparisons, all classes corresponding to overlaps were considered as a unified 'noise-class' and FP/FN errors were measured on the basis of the three main classes. In all cases, all the overlapping waveforms are accounted for to the total number of participating waveforms (1100 in our example).

For the neuron-based validation model, NASS is considered in its final resolution scheme. Overlaps are first resolved (Table 1C) and class populations are updated accordingly (Table 1C measurements are resolved in Table 2, section 'Detected'). The total number of participating waveforms is now modified to account for the total number of spikes fired by the three neurons ($3 \times 450 = 1350$ waveforms – see Table 2, section 'Fired').

Fig. 7(b) and (c) depicts the performance of all methods under comparison for both validation modes and using 10 different realisations of the two data sets (Dataset-1 and Dataset-2). Averaged values for the error rates and the FP–FN errors are shown as a function of the SNR level. For class-based validation (Fig. 7(b)), it can be seen that NASS achieves the lowest error rate for both data sets and this is lower than 2% when the SNR is higher than 3. The corresponding neuron-based validation mode for NASS can be seen in Fig. 7(c). For SNR values higher than 2, the error rate is lower than 10%. The corresponding performance of both other methods has also been included. The purpose was to demonstrate that the performance of traditional spike-sorting schemes drops (due to spike overlaps). For instance, the optimum class-based validation error rate of 0% for a class of Dataset-2 (exactly 300 correct single spikes classified under a class) corresponds to 11% error in terms of neuron-based validation; that is, 450 spikes were totally fired for a neuron in Dataset-2 with 150 of them being engaged in overlaps; this results in 150 FNs for this neuron (only 300 spikes were correctly classified), in a total of 1350 spikes.

3.4. Continuous time series simulated data

We applied NASS and Waveclus on publicly available continuous time series datasets: they were realistically compiled using extracted spike shapes from recordings in the neocortex and basal ganglia (Quian Quiroga et al., 2004). For the generation of background noise, over 500 different spike waveforms were employed; the noise level of the data set was determined based on its standard deviation (σ) and varied from 0.05 to 0.3, relatively to the amplitude of the spike classes. The firing of three neurons had been simulated

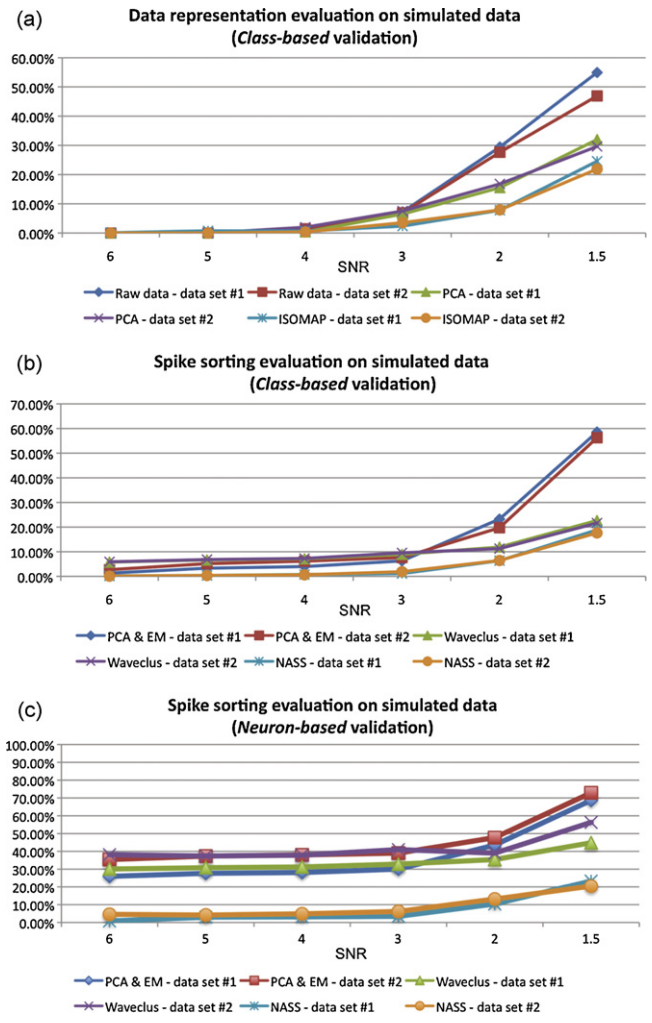

Fig. 7. Comparison of average error rates for both validation modes. (a and b) *Class-based*. (c) *Neuron-based*.

with mean firing rate of 20 Hz and a refractory period of 2 ms, while the inter-spike intervals followed a Poisson distribution. A short segment of the simulated trace, for $\sigma = 0.15$, is included in Fig. 8(a).

For consistency in the evaluation of both methods, the spike extraction step was implemented using Waveclus in both cases (NASS does not include any particular spike detection step); the evaluation followed the 'neuron-based' validation scheme. Each method classified the detected spikes into the corresponding neurons. Using the obtained classification labels, false-positive and negative errors were estimated, with respect to the known labels (ground-truth) indicating neuron's firing.

An example of sorted data for NASS and Waveclus is shown in Fig. 8(b1) and (b2), respectively. In the case of Waveclus, this is the final classification (it forces overlaps to be classified in one of the defined classes), while in the case of NASS, this classification is only an intermediate step. All the ambiguous waveforms (coloured in black) are directed, as described in Section 2.2 to side-branch for the final classification. A 2D ISOMAP representation space is shown in Fig. 8(c) (the optimal dimensionality r_0 was actually 4, but we restrict ourselves within two dimensions to ease presentation). Elements of the synthetic database are shown using circles (◯ for noisy and ● for noise-free waveforms) to denote double overlaps and stars (*) to denote single noisy spikes. The different overlapping combinations were denoted using three colours (magenta, green, red) while the single spikes are coloured blue. Ambiguous waveforms are shown using rectangles (□) for waveforms classified as overlaps and triangles (▷ △ ▽) for waveforms classified as single spikes (each for a separate class). All spikes under investigation are depicted using yellow as the filling colour, while the four colours (magenta, red, green and blue) mentioned above denote their classification. An exemplar of NASS classification results is shown in Fig. 8(d) using the same segment from Fig. 8(a) and the colour code used in Fig. 8(b). Prototype waveforms (as they were estimated by NASS) are used to approximate the real time-depended waveforms.

The detailed results including error rates, for every noise level, are shown in Table 3 for both methods. The initial error rate induced by the spike extraction step has also been estimated separately.

Table 2
Neuron-based validation mode: an example of results for NASS, derived for **Dataset-2**.

Templates	Fired		
	Neuron I	Neuron II	Neuron III
NASS			
Template I	300		
Template II		300	
Template III			300
Template I + II	50	50	
Template I + III	50		50
Template II + III		50	50
Template I + II + III	50	50	50
Totals per neuron	450	450	450
Total spikes		1350	
Templates	**Detected**		
	Neuron I	Neuron II	Neuron III
NASS			
Template I	300		
Template II		300	
Template III			300
Template I + II	50	50	
Template I + III	49	8	50
Template II + III	9	50	50
Template I:11 + 111	42	44	48
FPs per neuron	9	8	0
FNs per neuron	9	6	2
FPs + FNs per neuron	18	14	2
Total FPs + FNs		34	
Error rate		2.52%	

Table 3
Comparative evaluation of NASS and Waveclus using continuous time series simulated data sets. The evaluation followed the *neuron-based* validation scheme.

Data set		Extraction step			Waveclus			NASS				Δ_error	
σ (noise)	Fired	Detected	Missed	Extraction error	FP	FN	Error	Resolved	FP	FN	Error	Waveclus	NASS
0.05	3514	3123	391	11.13%	0	541	15.40%	3269	132	307	12.49%	4.27%	1.37%
0.10	3522	3164	358	10.16%	0	438	12.44%	3372	53	161	6.08%	2.27%	−4.09%
0.15	3477	2934	543	15.62%	0	610	17.54%	3103	40	398	12.60%	1.93%	−3.02%
0.20	3474	2547	927	26.68%	17	1044	30.54%	2881	67	875	27.12%	3.86%	0.43%
0.25	3298	1831	1467	44.48%	148	2029	66.01%	1978	165	1489	50.15%	21.53%	5.67%
0.30	3475	1250	2225	64.03%	28	2545	74.04%	1346	94	2242	67.22%	10.01%	3.19%

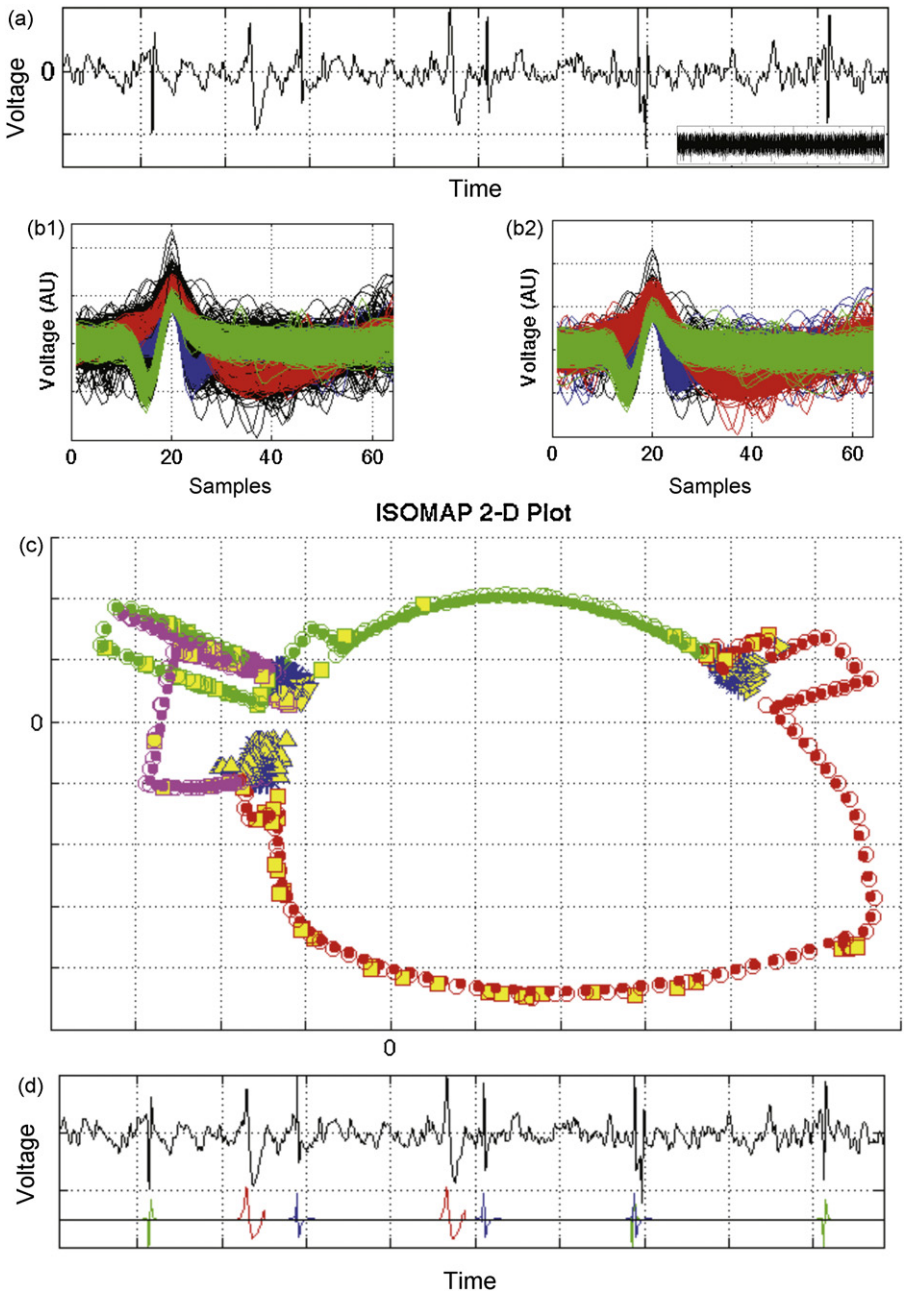

Fig. 8. Demonstration of NASS based on a realistic continuous time series ($\sigma = 0.15$); an amount of approximately 11% double overlaps was estimated, resulting from the cofiring of 22% spikes. (a) A typical segment (approximately 80 ms) of the simulated trace. (b1) Intermediate classification results for NASS; (b2) classification results for Waveclus. (c) 2D ISOMAP space used in the *side-branch* processing; spikes from the synthetic database and ambiguous spikes left unclassified in the previous step are shown (for details Section 3.4). (d) Classification results are showed for segment originally presented in (a).

Based on this estimation, a relative Δ_error has been assigned to both methods. It can be observed that in medium noise levels the Δ_error for NASS is negative. This reflects the ability of NASS to resolve overlaps and thus improve the overall performance of the spike-sorting task.

4. Discussion

The present work introduces NASS, an offline method for spike sorting that successfully addresses both contemporary challenges: noise and spike overlaps. NASS combines the robustness of 'data manifold' representation (ISOMAP) with the efficiency of novel schemes in learning theory (ELM). It is an empirical approach in that the few necessary parameters (number of active neurons and complexity of overlaps) can be provided, readily, by the user after a short interaction with low-dimensional scatterplots produced at the first stage of processing. While the first stage of processing shares commonalities with many traditional spike-sorting routines (i.e., a clustering step assigns waveforms into distinct neurons), the engagement of 'principled learning' during the second stage results in a very novel approach, which can efficiently handle the overlaps.

The main advantage of NASS lies within the 'side branch'. Avoiding any theoretical modelling for the noise profile, it empirically fits to the actual data. The use of inherent noise constitutes NASS capable of resolving, within a single step, the ambiguous noisy waveforms (left unclassified during the first step) into either noisy single spikes or overlaps of known origin.

Alternative approaches to overlap resolution do exist in related literature. The most popular method is template matching (Takahashi et al., 2003; Zhang et al., 2004; Vargas-Irwin and Donoghue, 2007; Ding and Yuan, 2008; Franke et al., 2009). However, template subtraction methods suffer from the inability to cope with correlated template waveforms (Franke et al., 2009). These methods follow a 'top-down' approach and attempt a sequential decomposition that involves multiple threshold decision steps.

On the contrary, NASS adopts a 'bottom-up' approach. Individual prototypes are initially derived and then variably combined, partially coupled with noise extracts from the recording. This synthetic database and the ambiguous waveforms in the recording are then represented in a low-dimensional ISOMAP space. At the top level, in a single classification problem, ELM searches the best fit for every ambiguous waveform among the compiled database, using their ISOMAP coordinates. Thus, intermediate errors are avoided.

Beyond template subtraction, there is a restricted number of methods (Wang et al., 2006; Herbst et al., 2008) aiming at resolving spike overlaps as well. They are built over different principles and therefore are difficult to be directly compared. For instance, the method of Wang et al. works in the frequency domain and uses phase information in an elegant way to decompose a complex waveform into its constituents. The authors measured performance using an index of 'correct classification' (the number of correctly classified spikes, divided by the total number of spikes) that incorporates FNs for each class, but ignores the corresponding FPs. Following a different spirit, the method of Herbst et al. is based on Markov-chain modelling and – in theory – can incorporate overlaps of high complexity. However, the computational cost is apparently intractable. On the contrary, our approach – by exploiting the efficiency of ELM – scales up reasonably well. Based on user's knowledge regarding the nature of experimental data (see Section 3.1), it can be tailored to handle high-complexity overlaps.

The introduced framework can be optimised in many ways. Actually, some of the improvements could have been accommodated directly. They were left out though, to keep this introduction as simple as possible. For instance, the relative firing probability of the detected neurons can be easily deduced (from the populations within the FCM-based groups) and used to build the synthetic database to contain the corresponding overlaps at the right portions (instead of the uniform/exhaustive fashion implied in Section 2.2).

Several open issues in spike sorting such as neuronal instability, reduced spike size due to bursting or low frequency oscillations in the recording signal due to external sources (e.g., breathing and blood flow), have not been dealt in the present study. In general, these situations introduce non-Gaussian components in the spike shape variance (Quian Quiroga et al., 2004) and raise significant difficulties to algorithms that assume a particular noise distribution. Thus, the use of algorithms like the one we are introducing here, which do not make statistical assumptions about the data, may be favourable. One of the main aims in future research will be the application of NASS in datasets presenting such challenges.

Finally, one direction that deserves additional study is the further development of the NASS-scheme in a fully automated technique. Some emerging ideas from the field of stability-based clustering seem suitable to accomplish this task (Lange et al., 2004; Shamir and Tishby, 2007). By incorporating them at the end of the main processing stage, the actual number of underlying neurons can be deduced automatically.

Appendix A. The ISOMAP algorithm

Isometric feature mapping technique (ISOMAP) uses local metric information to characterise the underlying global geometry of a data set (Tenenbaum et al., 2000). ISOMAP is an extension of classical MDS that includes a transformation of the original distance matrix $D_{[N\times N]}$ to the matrix $GD = G(D)$ that contains the shortest path distance between all pair of points.

Step I. A weighted graph G is defined over all N points by connecting points P_i and P_j if P_i is among the k-nearest neighbours of P_j (or conversely). The corresponding edge weights are initialised to $GD(i,j) = D(i,j)$ if P_i, P_j are linked by an edge; $GD(i,j) = \infty$ otherwise.
Step II. For each $k = 1, 2, \ldots, N$ in turn, all entries $GD(i,j)$ are replaced by $\min\{GD(i,j), GD(i,k) + GD(j,k)\}$. The fraction of points not connected to the main component of the resulting graph is detected and deleted from further analysis. As ε is reduced, more points are deleted.
Step III. The images Y_i of points P_i in a space of reduced dimensions r are derived via the application of classical MDS, $\mathbf{Y}_{[N'\times r]} = \mathrm{MDS}_r(\mathbf{GD})$.

One important issue of the dimensionality reduction method is determination of the dimensionality r of the reduced space. The ISOMAP algorithm computes the residual variance (RV), to evaluate the error of dimensionality reduction: $\mathrm{RV} = 1 - R^2$ (**DG**, **DY**), where **DY** is the Euclidean distance matrix in the derived r–D space and R is the standard linear correlation coefficient, taken over all entries of **DG** and **DY**.

Appendix B. Extreme Learning Machine (ELM)

ELM employs a very simple learning algorithm which, based on N distinct input–output pairs $(\mathbf{x}_j, \mathbf{t}_j)$ with $\mathbf{x}_j \in R^n$ and $\mathbf{t}_j \in R^m$, designs a single-layer forward network (SLFN), that reaches the smallest training error while achieving high generalisation performance (Huang et al., 2006).

Following the standard convention, the output of the network is written as

$$\sum_{i=1}^{L} \boldsymbol{\beta}_\mathbf{i} g(\mathbf{w}_i \cdot \mathbf{x}_j + b_i) = \mathbf{o}_\mathbf{j}, \quad j = 1, \ldots, N$$

with L a user-defined number ($L \ll N$) of hidden neurons, $g(x)$ a user-defined type of activation function (e.g., sigmoid), $\boldsymbol{\beta}_\mathbf{i} = [\beta_{i1}, \beta_{i2}, \ldots, \beta_{im}]$ the weight vector connecting the ith hidden node with the output nodes, b_i the threshold of the corresponding hidden node and $(\mathbf{w}_i \cdot \mathbf{x}_j)$ denoting the inner-product of $\mathbf{w}_i$ and $\mathbf{x}_j$. The ELM algorithm can then be summarised within the following steps.

Step 1. Randomly assign $\mathbf{w}_i$ and b_i.
Step 2. Calculate the hidden layer output matrix $\mathbf{H}_\mathbf{L}$:

$$\mathbf{H}_{\mathbf{L}[N\times L]} = \mathbf{H}(L) = \begin{bmatrix} g(\mathbf{w}_1 \cdot \mathbf{x}_1 + b_1) & \cdots & g(\mathbf{w}_\mathbf{L} \cdot \mathbf{x}_1 + b_L) \\ & \cdots & \\ g(\mathbf{w}_1 \cdot \mathbf{x}_N + b_1) & \cdots & g(\mathbf{w}_\mathbf{L} \cdot \mathbf{x}_N + b_L) \end{bmatrix}$$

Step 3. Calculate the output weight matrix:

$$\boldsymbol{B} = \begin{bmatrix} \boldsymbol{\beta_1} \\ \vdots \\ \boldsymbol{\beta_L} \end{bmatrix} = \boldsymbol{H}_L^* \boldsymbol{T} = \boldsymbol{H}_L^* \begin{bmatrix} \boldsymbol{t_1} \\ \vdots \\ \boldsymbol{t_N} \end{bmatrix}$$

where $\mathbf{H}^*$ denotes the Moore–Penrose generalised inverse of matrix $\mathbf{H}$.

When ELM algorithm is used to craft the SLFN for a classification task, the output layer contains as many neurons as the number of classes in the data (Huang, 2009). In this case the training data, which are originally in the form of pairs $\{(\mathbf{x}_j, \mathrm{t}_j)\}_{j=1:N}$ with

$t_j \in \{1, \ldots, m\}$, are transformed to $\{(\mathbf{x}_j, \mathbf{t}_j)\}_{j=1:N}$. The vectors $\mathbf{t}_j$ are ordered m-tuples, which encode the classification label. They contain an '1' at the place corresponding to the class-label integer and zeros elsewhere.

References

Abeles M, Goldstein JM. Multi-spike train analysis. Proc IEEE 1977;65:762–73.

Adamos DA, Kosmidis EK, Theophilidis G. Performance evaluation of PCA-based spike sorting algorithms. Comput Methods Prog Biol 2008;91(3):232–44.

Bar-Gad I, Ritov Y, Vaadia E, Bergman H. Failure in identification of overlapping spikes from multiple neuron activity causes artificial correlations. J Neurosci Methods 2001;107(1–2):1–13.

Ding W, Yuan J. Spike sorting based on multi-class support vector machine with superposition resolution. Med Biol Eng Comput 2008;46(2):139–45.

Fee MS, Mitra PP, Kleinfeld D. Variability of extracellular spike waveforms of cortical neurons. J Neurophysiol 1996a;76(6):3823–33.

Fee MS, Mitra PP, Kleinfeld D. Automatic sorting of multiple unit neuronal signals in the presence of anisotropic and non-Gaussian variability. J Neurosci Methods 1996b;69(2):175–88.

Franke F, Natora M, Boucsein C, Munk MH, Obermayer K. An online spike detection and spike classification algorithm capable of instantaneous resolution of overlapping spikes. J Comput Neurosci 2009 [Epub ahead].

Glaser EM, Marks WB. On-line separation of interleaved neuronal pulse sequences. Data Acquis Process Biol Med 1968;5:137–56.

Harris KD, Henze DA, Csicsvari J, Hirase H, Buzsaki G. Accuracy of tetrode spike separation as determined by simultaneous intracellular and extracellular measurements. J Neurophysiol 2000;84(1):401–14.

Herbst JA, Gammeter S, Ferrero D, Hahnloser RH. Spike sorting with hidden Markov models. J Neurosci Methods 2008;174(1):126–34.

Huang GB. MATLAB codes of ELM algorithm; 2009, http://www3.ntu.edu.sg/home/egbhuang/ELM_Codes.htm.

Huang GB, Zhu QY, Siew CK. Extreme learning machine: theory and applications. Neurocomputing 2006;70(1–3):489–501.

Hulata E, Segev R, Ben-Jacob E. A method for spike sorting and detection based on wavelet packets and Shannon's mutual information. J Neurosci Methods 2002;117(1):1–12.

Kim J, Shin HS, Shin K, Lee M. Robust algorithm for arrhythmia classification in ECG using extreme learning machine. Biomed Eng Online 2009;8:31.

Lange T, Roth V, Braun ML, Buhmann JM. Stability-based validation of clustering solutions. Neural Comput 2004;16(6):1299–323.

Laskaris NA, Ioannides AA. Semantic geodesic maps: a unifying geometrical approach for studying the structure and dynamics of single trial evoked responses. Clin Neurophysiol 2002;113(8):1209–26.

Laskaris NA, Fotopoulos S, Ioannides AA. Mining information from event-related recordings. IEEE Signal Proc Mag 2004;21(3):66–77.

Laskaris NA, Zafeiriou SP, Beyond FCM. Graph-theoretic post-processing algorithms for learning and representing the data structure. Pattern Recogn 2008;41(8):2630–44.

Letelier JC, Weber PP. Spike sorting based on discrete wavelet transform coefficients. J Neurosci Methods 2000;101(2):93–106.

Lewicki MS. A review of methods for spike sorting: the detection and classification of neural action potentials. Network: Comput Neural Syst 1998;9:R53–78.

Pouzat C, Mazor O, Laurent G. Using noise signature to optimize spike-sorting and to assess neuronal classification quality. J Neurosci Methods 2002;122(1):43–57.

Quian Quiroga R, Nadasdy Z, Ben-Shaul Y. Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering. Neural Comput 2004;16(8):1661–87.

Samko O, Marshall AD, Rosin PL. Selection of the optimal parameter value for the ISOMAP algorithm. Pattern Recogn Lett 2006;27(9):968–79.

Shamir O, Tishby N. Cluster stability for finite samples. Adv Neurol 2007.

Takahashi S, Anzai Y, Sakurai Y. Automatic sorting for multi-neuronal activity recorded with tetrodes in the presence of overlapping spikes. J Neurophysiol 2003;89(4):2245–58.

Tenenbaum JB, de Silva V, Langford JC. A global geometric framework for nonlinear dimensionality reduction. Science 2000;290(5500):2319–23.

Vargas-Irwin C, Donoghue JP. Automated spike sorting using density grid contour clustering and subtractive waveform decomposition. J Neurosci Methods 2007;164(1):1–18.

Wang GL, Zhou Y, Chen AH, Zhang PM, Liang PJ. A robust method for spike sorting with automatic overlap decomposition. IEEE Trans Biomed Eng 2006;53(6):1195–8.

Wehr M, Pezaris JS, Sahani M. Simultaneous paired intracellular and tetrode recordings for evaluating the performance of spike sorting algorithms. Neurocomputing 1999;26–27:1061–8.

Zafeiridou G, Theophilidis G. The action of the insecticide imidacloprid on the respiratory rhythm of an insect: the beetle *Tenebrio molitor*. Neurosci Lett 2004;365(3):205–9.

Zhang PM, Wu JY, Zhou Y, Liang PJ, Yuan JQ. Spike sorting based on automatic template reconstruction with a partial solution to the overlapping problem. J Neurosci Methods 2004;135(1–2):55–65.

Zigkolis CN, Laskaris NA. Using conditional FCM to mine event-related brain dynamics. Comp Biol Med 2009;39(4):346–54.

Zouridakis G, Tam DC. Identification of reliable spike templates in multi-unit extracellular recordings using fuzzy clustering. Comput Methods Prog Biol 2000;61(2):91–8.