Theoretical Studies and Algorithms Regarding the Solution of Non-invertible Nonlinear Source Separation

David Felice, Rafael Assato Ando, Leonardo Tomazeli Duarte, Christian Jutten, R Attux

Theoretical Studies and Algorithms Regarding the Solution of Non-Invertible Nonlinear Source Separation

D. F. F. Baptista[1], R. A. Ando[1,2], L. T. Duarte[3], C. Jutten[2], R. Attux[1⋆]

[1] School of Electrical and Computer Engineering, UNICAMP, Brazil
[2] GIPSA-Lab, Université Joseph Fourier (UJF), France
[3] School of Applied Sciences, UNICAMP, Brazil

ABSTRACT

In this paper, we analyse and solve a source separation problem based on a mixing model that is nonlinear and non-invertible at the space of mixtures. The model is relevant considering it may represent the data obtained from ion-selective electrode arrays. We apply a new approach for solving the problems of local stability of the recurrent network previously used in the literature, which allows for a wider range of source concentration. In order to achieve this, we utilize a second-order recurrent network which can be shown to be locally stable for all solutions. Using this new network and the priors that chemical sources are continuous and smooth, our proposal performs better than the previous approach.

1 PROBLEM STATEMENT

The general blind source separation (BSS) problem consists in estimating sources, represented by the vector $\mathbf{s} = [s_1, s_2, ..., s_n]^T$, that have been mixed by an unknown function $\mathcal{F}(.)$, given only the mixtures $\mathbf{x} = [x_1, x_2, ..., x_n]^T$ and prior information on the model or the sources.

$$\mathbf{x} = \mathcal{F}(\mathbf{s}). \tag{1}$$

For a linear function $\mathcal{F}(.)$, the problem can be uniquely solved – up to scale and permutation ambiguities – by formulating a criterion of statistical independence, but this is no longer possible for a generic nonlinear mapping. [2, 5, 6]. It is well known that the nonlinear BSS problem is very difficult to solve, since the generic nonlinearities can cause multiple statistical independent solutions that are still mixtures of the sources.

In the literature, we can find several approaches for dealing with specific nonlinear mixtures, such as the Post-Nonlinear (PNL) [2, 7] and the Linear-Quadratic (LQ) [4]. For the latter, a recurrent network has been proposed as

⋆ The authors would like to thank FAPESP, CNPq, CNRS and ERC project 2012-ERC-AdG-320684 CHESS for the funding.

part of the solving system, and, with some modifications, can also be used for a different mixing model which shall be analyzed in this paper, represented as follows:

$$\begin{aligned} x_1 &= s_1 + a_1 s_2^2 \\ x_2 &= s_2 + a_2\sqrt{s_1} \end{aligned} \tag{2}$$

This model describes data obtained from ion-selective-electrodes (ISEs), where we have two sources (i.e. chemical species) and two sensors. More details about the suitability of model (2) to the problem can be found in [3].

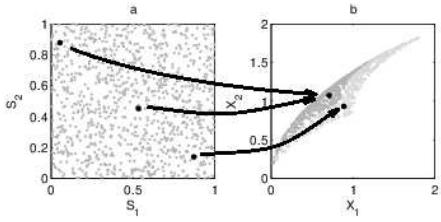

Fig. 1. Illustration of the folding of the source space

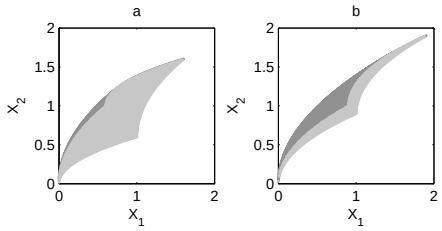

Fig. 2. Increase of the folded area with the selectivity coefficients

Our main goal in this paper is to obtain an estimate $\mathbf{y} = (y_1, y_2)$ of the sources $\mathbf{s} = (s_1, s_2)$, when the exact selectivity coefficients a_{ij} and the mixtures $\mathbf{x} = (x_1, x_2)$ are known. For the mixing model (2), the nonlinearity creates a difficulty for being non-invertible, since two distinct source points can be mapped onto the same mixture. Indeed, as can be seen in Fig. 1, the model effectively "folds" the source space, and this folding depends on the selectivity coefficients, as seen in Fig. 2. For a given point in the mixture space $\mathbf{x} = (x_1, x_2)$, it can be seen that the possible solutions are the sources $\mathbf{y} = (s_1, s_2)$ and a mixture of the sources, given by:

$$\mathbf{y}^* = \left(\frac{\left(\sqrt{s_1}\left(a_1 a_2^2 - 1\right) + 2a_1 a_2 s_2\right)^2}{\left(a_1 a_2^2 + 1\right)^2}, \frac{-a_1 a_2^2 s_2 + 2a_2\sqrt{s_1} + s_2}{a_1 a_2^2 + 1} \right) \tag{3}$$

The folding frontier (i.e., the *locus* of the points for which $\mathbf{y} = \mathbf{y}^*$) is given by:

$$a_1 a_2 s_2 = \sqrt{s_1}. \tag{4}$$

Since we are dealing with aqueous ionic solutions, we can restrict the sources concentrations to $s_i \in [0, 1]$. In this region of interest, we shall henceforth call the points below and above the folding frontier regions 1 and 2 respectively, as seen in Fig. 3. It can also be seen that while some of the points in region 1 are mapped to an invertible area (i.e., $\mathbf{y}^*$ is outside the region of interest, and therefore no ambiguity ensues), all points from region 2 are mapped to the non-invertible area.

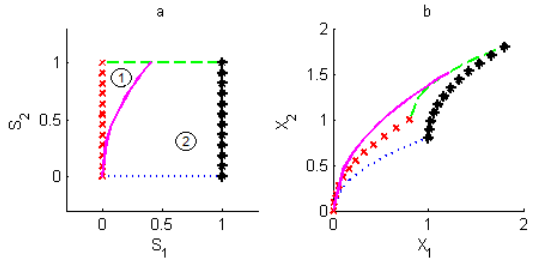

Fig. 3. Mapping of the frontiers in the source and mixture spaces

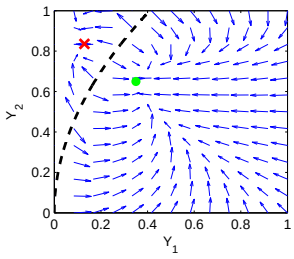

Fig. 4. Direction of network step based on the initialization point

2 RECURRENT NETWORK ANALYSIS

The use of recurrent networks to solve this problem has already been considered before by Duarte and Jutten [3], who achieved good but limited results since the network used is not locally stable for all possible ionic concentrations. In order to solve this issue, a new approach that implements a second-order recurrent network based on the Newton-Raphson's method has been proposed [1]. The goal in using a recurrent network is to obtain the solutions of the model (2) numerically, since obtaining the analytical solution is not always straightforward.

2.1 Recurrent Network

The main idea to obtain a separating system is to represent (2) as

$$\mathbf{G}(\mathbf{s}) = \begin{bmatrix} s_1 + a_1 s_2^2 - x_1 \\ s_2 + a_2\sqrt{s_1} - x_2 \end{bmatrix} = \mathbf{0} \tag{5}$$

Then, the source separation problem can be interpreted as a homogeneous non-linear equation system and can be solved by root-finding algorithms such as the Newton-Raphson's method [1]. This leads to the following equation:

$$\mathbf{y}(m+1) = \mathbf{y}(m) - \mu {\mathbf{J}_\mathbf{G}}^{-1} \mathbf{G}(\mathbf{y}(m)) \tag{6}$$

where $\mathbf{y}$ is the estimate of the sources, $\mu \in (0, 1]$ is an adjustment scale factor, m is the iteration index of the network and $\mathbf{J_G}$ is the Jacobian matrix of $\mathbf{G}$. To avoid discontinuities in the derivatives when extending (2) to $\mathbb{R}^2$, we can use:

$$\begin{aligned} x_1 &= y_1 + a_1 y_2^2 \\ x_2 &= y_2 + a_2\sqrt{|y_1|}\text{sign}(y_1) \end{aligned} \tag{7}$$

which is the same model as (2) in the region of interest $[0, 1] \times [0, 1]$.

When the network converges, we obtain $\mathbf{y}(m+1) = \mathbf{y}(m)$, which is defined as a fixed point of the network. From (6), we can show that this happens if and only if $\mathbf{G} = \mathbf{0}$, since ${\mathbf{J}_\mathbf{G}}^{-1}$ is non-singular.

2.2 Stability of Solutions

We can calculate the stability condition of Eq. (6), for all fixed points. It is well known that for discrete dynamic systems, if all eigenvalues λ_i of the Jacobian matrix of the recurrence evaluated at the fixed point satisfy $|\lambda_i| < 1$, the system is locally stable. Computing the Jacobian matrix at the fixed points, leads to:

$$\mathbf{J}|_{\mathbf{G=0}} = \begin{bmatrix} \frac{\partial y_1(m+1)}{\partial y_1(m)} & \frac{\partial y_1(m+1)}{\partial y_2(m)} \\ \frac{\partial y_2(m+1)}{\partial y_1(m)} & \frac{\partial y_2(m+1)}{\partial y_2(m)} \end{bmatrix}_{\mathbf{G=0}} = \begin{bmatrix} 1-\mu & 0 \\ 0 & 1-\mu \end{bmatrix} \tag{8}$$

From (8) we conclude that for $\mu \in (0, 1]$, the fixed points are always locally stable. According to this result, the stability problem found in [3] no longer exists, since the solution (s_1, s_2) is stable for all source concentrations. However, we now have to deal with a non-separating solution (3) that is also stable. Since the two solutions are always in different regions, it is important to know how to control which solution we want the network to obtain. Given a sufficiently small step for the network in (6) – e.g., $\mu = 0.1$ –, if we initialize in $(0.9, 0.1)$ and $(0.1, 0.9)$ we converge to solutions in regions 1 and 2 respectively, as seen in Fig. 4. As a result, specifying the initial point can be used to reach the separating solution of the model.

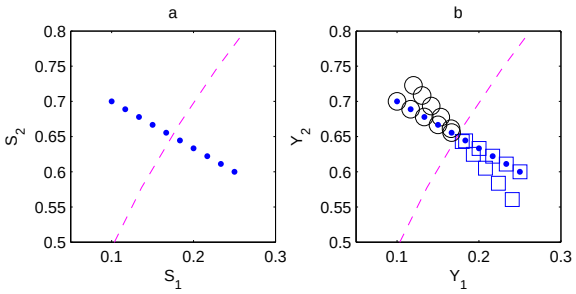

Fig. 5. Trajectories near the folding frontier.

2.3 Pivot Points

To have more knowledge about the solutions' trajectories near the folding frontier, we can look at Fig. 5. In Fig. 5a, we see the trajectory of the sources when it crosses the frontier. In Fig. 5b, we see two different trajectories in the (y_1, y_2) plane: the one represented by squares, corresponding to the estimate when we initialize in region 1, and one represented by circles when we initialize in region 2.

From the initial point of the trajectory, we can see that the solution corresponds to the square estimate before crossing the frontier, and the circle one

afterwards. Let us define the point when the solution crosses the frontier as the *pivot*. Moreover, the *pivot* is also the point at which the initialization should be changed.

After observing the behaviour of the trajectory of the sources and the estimates in several simulations, it was suggested that the correct solution should combine the estimates in such a way that the resulting signal would be as smooth as possible near the folding frontier. As can also be seen in Fig. 5, we can verify that, near the border, the correct solution is smoother than the alternative in which no change of initialization occurs. Therefore, using as priors that the source are continuous and smooth, we propose an algorithm that determines which initialization should be applied to the points at the non-bijective region in order to obtain the correct solution.

3 PROPOSED ALGORITHM

In this section, we propose an algorithm that identifies the pivots and the initializations that should be used to recover the signal. In the following sections, we will present the technical aspects of the proposed algorithm.

3.1 Identification of Potential Pivots

In order to identify the pivots, we calculate the entire estimate with only the initialization in region 1 and make a system that predicts, at a given point of the calculated signal, if the next point would cross the border. The prediction does not have to be very accurate, since it only identifies potential pivots. After identifying all pivots, we segment the signal into blocks in which all the points are either initialized in region 1 or 2. In Fig. 6 we can see two signals in each graph, a thin one representing the estimated sources obtained with initialization at region 1, and a thick one representing the real sources. The vertical lines represent potential pivots given by the algorithm, and as we can see, all points where the initialization should be changed were identified.

For the prediction of the next point, a factor based on the curvature of the trajectory of the estimates was included. This is because when the sources cross the border, the trajectories of the estimates tend to rapidly change their direction, yielding high curvature. The algorithm therefore considers such abrupt variations as an indicative that the next point is a probable pivot. For more details about how the prediction is done, one can look at the pseudocode in Table 1.

3.2 Classification of each Block

After segmenting the signal, we need to identify which initialization each block should be given. For that, we initially check for points outside the non-bijective area. As previously mentioned, we know that such points can only come from region 1, so we can safely initialize its corresponding block in it. For the remaining

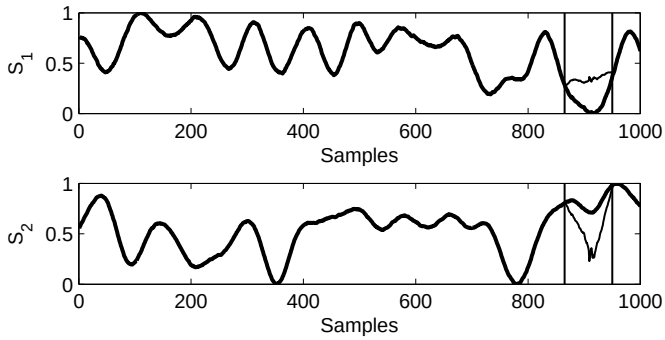

Fig. 6. Block segmentation of the estimates. Thick and thin curves represent the sources and the estimate, respectively.

blocks, we select the initialization that maximizes the measure of smoothness around the pivot. We define smoothness as:

$$S(\mathbf{y}) = \mathrm{Var}(\mathbf{y}^2(n+1) - \mathbf{y}^2(n)) \tag{9}$$

where Var(.) is the empirical variance calculated in a N-sized window ($N = 11$ samples) centered around the pivot and n is the sample index of the signal. A pseudocode of the entire algorithm can be seen in Table 1:

Table 1. Pseudocode of the proposed algorithm

1. Using (6), obtain $\mathbf{y}$, the estimate from initialization point 1.
2. For each point of $\mathbf{y}$, predict the next point using:

$$\hat{\mathbf{y}}(n+1) = \mathbf{y}(n) + \lambda(n)\frac{\mathbf{\Delta y}(n)}{||\mathbf{\Delta y}(n)||} \tag{10}$$

$$\lambda(n) = \frac{\sum_{i=n-4}^{n} ||\mathbf{\Delta y}(i)||}{5} \exp(1 - |cos\theta|) \tag{11}$$

   where $\mathbf{\Delta y}(n) = \mathbf{y}(n) - \mathbf{y}(n-1)$ is the last step, $\theta(n)$ the angle between $\mathbf{\Delta y}(n+1)$ and $\mathbf{\Delta y}(n+3)$, and $\lambda(n)$ a scalar factor based on the curvature of the trajectory.
3. If $\hat{\mathbf{y}}(n+1)$ is in region 2, store it as a potential pivot.
4. Segment signal into blocks separated by potential pivots.
5. Assign non-ambiguous blocks to region 1.
6. For the remaining blocks, test initializations in both regions and select the combination that minimizes the smoothness (9).

4 SIMULATION RESULTS

In this section, we analyse the performance of the proposed algorithm and compare it to the previous methods for solving the problem [3]. We simulate sources by filtering a uniform random signal with a high order low-pass filter, with variable cut-off frequency. As one may notice, the smaller the cut-off frequency, the smoother the estimated sources obtained. To estimate the performance of the technique, we measure the Signal-to-Interference ratio (SIR) – defined in Eq. (12) – for each block of the signal.

$$\mathrm{SIR} = 10 \log_{10} \left(\frac{E\{\mathbf{s}^2\}}{E\{(\mathbf{s}-\mathbf{y})^2\}} \right) \tag{12}$$

If a block's SIR is lower than a certain threshold, we consider that the estimates are wrong, otherwise we consider them correct. For our simulation, we used 15 dB as the threshold. We can then calculate the percentage of errors given by the algorithm.

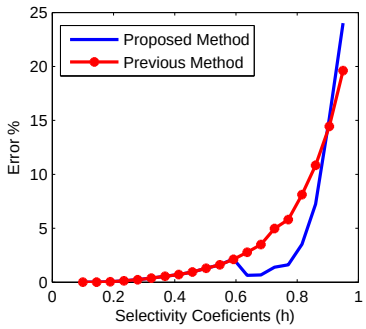

Fig. 7. Simulation results for fixed source smoothness and varying selectivity coefficient

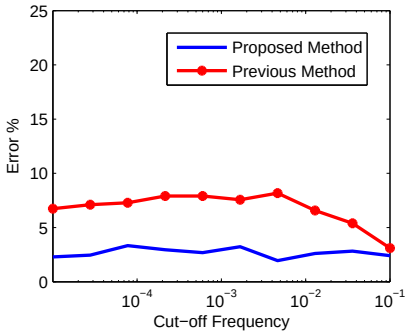

Fig. 8. Simulation results for fixed selectivity coefficient ($h = 0.8$) and varying source smoothness

4.1 First Scenario

In this simulation, we keep the sources' smoothness constant and vary the selectivity coefficients h, defined, for the sake of illustration, as $a_1 = a_2 = h$. Fig. 7 shows the average percentage of errors, for 300 trials for each h and 20 different $h \in [0.1, 0.95]$. We can verify that for selectivity coefficients $h < 0.6$, results for both the proposed network and the previous one can be considered similar. However, for $h \in [0.6, 0.9]$ our approach yields considerably better results. For $h > 0.9$ (i.e. for ill-conditionned mixtures), the errors with the two methods are above 10% and grow rapidly with h, and we can consider that both methods fail.

4.2 Second Scenario

In the second scenario, we keep the selectivity coefficients constant ($h = 0.8$) and vary the sources' smoothness by adjusting the filter's cutoff frequency. As we can see in Fig. 8, for all cutoff frequencies tested, the proposed algorithm performed better than the previous technique.

We can also see that regardless of the signal's smoothness, the percentage of errors in our technique remains roughly constant and very low (about 2%), which suggests the smoothness of the sources do not interfere in the algorithm's performance.

5 CONCLUSION

In this paper we considered a problem of source separation in non-invertible nonlinear mixtures derived from an application in which chemical sensor arrays are used to measure ionic concentrations. For the simple case of two mixtures and two sources, we proposed a new method for solving the problem which has better stability properties. The nonlinear mapping studied presented difficulties caused by the existence of multiple solutions, which makes the model non-invertible. Nonetheless, our proposed method was capable of solving it using the prior source smoothness, and experimental results attested to the efficiency of the method even when the mixture is ill-conditioned.

Future works include extending the method to the blind case (i.e. when the mixing coefficients are unknown) and investigating other approaches, such as how the overdetermined scenario (i.e., with more sensors than sources), could provide additional information which would help solve the non-invertibility problem.

References

1. Ando, R.A.: Blind Source Separation in the Context of Polynomial Mixtures. Master's thesis, School of Electrical and Computer Engeneering, University of Campinas (2013)
2. Comon, P., Jutten, C.: Handbook of Blind Source Separation: Independent component analysis and applications. Academic press (2010)
3. Duarte, L.T., Jutten, C.: Design of smart ion-selective electrode arrays based on source separation through nonlinear independent component analysis. Oil & Gas Science and Technology–Revue d'IFP Energies nouvelles 69(2), 293–306 (2014)
4. Hosseini, S., Deville, Y.: Blind separation of linear-quadratic mixtures of real sources using a recurrent structure. In: Artificial Neural Nets Problem Solving Methods, pp. 241–248. Springer (2003)
5. Hyvärinen, A., Pajunen, P.: Nonlinear independent component analysis: Existence and uniqueness results. Neural Networks 12(3), 429–439 (1999)
6. Jutten, C., Karhunen, J.: Advances in blind source separation (bss) and independent component analysis (ica) for nonlinear mixtures. International Journal of Neural Systems 14(05), 267–292 (2004)
7. Taleb, A., Jutten, C.: Source separation in post-nonlinear mixtures. Signal Processing, IEEE Transactions on 47(10), 2807–2820 (1999)