SoftwareX 8 (2018) 26–32

Contents lists available at ScienceDirect

SoftwareX

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

Original software publication

A new algorithm in singular spectrum analysis framework: The Overlap-SSA (ov-SSA)

M.C.R. Leles [a,b,*], J.P.H. Sansão [a,b], L.A. Mozelli [a,d], H.N. Guimarães [c]

[a] CELTA — Center for Studies in Electronics Engineering and Automation, UFSJ — Federal University of São João del-Rei, Rod. MG 443 km 7, 36420-000, Ouro Branco, MG, Brazil
[b] PPGEE — Graduate Program in Electrical Engineering, UFMG — Federal University of Minas Gerais, Av. Antônio Carlos 6627, 31270-901, Belo Horizonte, MG, Brazil
[c] Department of Electrical Engineering-UFMG, Brazil
[d] Department of Electronics Engineering-UFMG, Brazil

ARTICLE INFO

Article history:
Received 6 November 2017
Accepted 6 November 2017

Keywords:
Singular spectrum analysis
Non-stationary signals
Segmentation

ABSTRACT

The Singular Spectrum Analysis (SSA) is powerful method, capable of working with arbitrary statistical process and it is adaptive to the underlaying data. Many variations of the standard methodology have been prosed in recent years improving the performance, adjusting to specific problems or objectives, or addressing some shortcomings. One of such drawbacks occurs when the spectrum spreads and varies over time, demanding many elementary matrices to reconstruct an approximation of the original series, hampering the method applicability. Also, another difficulty arises when large datasets are analyzed. There are computational issues and also problems with the method ability to maintain satisfactory separability. To circumvent these issues, a new method has been proposed. The original time series is divided into smaller and consecutive segments, with some superposition between them. Then, standard SSA is applied to each segment and the results are concatenated properly. This paper provides an implementation of this algorithm and some experiments are shown to illustrate the improvements achieved.

Code metadata

Current code version	v1.0
Permanent link to code/repository used for this code version	https://github.com/ElsevierSoftwareX/SOFTX-D-17-00081
Legal Code License	The 2-Clause BSD License
Code versioning system used	none
Software code languages, tools, and services used	MATLAB R2014a
Compilation requirements, operating environments &	MATLAB, Windows, Mac OS X, Linux
If available Link to developer documentation/manual	
Support email for questions	mleles@ufsj.edu.br

1. Motivation and significance

Singular Spectrum Analysis (SSA) is a growing methodology for time-series analysis that has been successfully applied in several fields of science and engineering [1,2].

Standard SSA algorithm consists of four stages: embedding, Singular Value Decomposition (SVD), grouping and diagonal averaging. In this way, SSA is capable of decomposing a time-series into

---

DOI of original article: http://dx.doi.org/10.1016/j.dsp.2017.10.025.

[*] Corresponding author at: CELTA — Center for Studies in Electronics Engineering and Automation, UFSJ — Federal University of São João del-Rei, Rod. MG 443 km 7, 36420-000, Ouro Branco, MG, Brazil.

E-mail addresses: mleles@ufsj.edu.br (M.C.R. Leles), joao@ufsj.edu.br (J.P.H. Sansão), mozelli@cpdee.ufmg.br (L.A. Mozelli), hng@cpdee.ufmg.br (H.N. Guimarães).

https://doi.org/10.1016/j.softx.2017.11.001

main components: trend, oscillations and noise. A great advantage is the fact that the methodology is nonparametric, meaning that it can adapt itself to the underlaying dataset, dismissing the necessity of *a priori* models. For this reason is also known as a model-free approach.

Despite recent success, there is room for improvement concerning the standard SSA algorithm. Due to high computational cost associated with SVD, its application to large datasets may not be suitable. To overcome this issue Korobeynikov [3] proposed a more efficient implementation of the SVD stage. Another approach has been proposed by Rekapalli and Tiwari [4], which consists in the segmentation of the original dataset into non overlapping segments, called Windowed SSA (WSSA).

Segmentation is a recurrent procedure in signal processing of non-stationary time-series [5]. Besides the reduction in computational cost, segmentation can provide more advantages, such as the improvement in time-frequency characterization of the analyzed signal [6]. Towards this end, a local version of the SSA, called Multi-Scale SSA (MS-SSA), has been proposed in Yiou et al. [7]. For a given stretch of the original dataset, the MS-SSA algorithm runs standard SSA in several segments of different lengths with coincident centers. It seeks information in distinct time-scale resolutions, similar to the wavelet transform.

Although the methodology proposed in Rekapalli and Tiwari [4] had lessen the computational effort in comparison with the standard SSA, there is no treatment for the boundary effect that arises at the union of consecutive segments. This issue may cause severe discontinuities in the reconstructed signal, introducing undesirable artifacts. On the other hand, the methodology proposed in Yiou et al. [7] improves the time-frequency characterization and signal reconstruction compared with the standard SSA, but the impact of several runs of the standard SSA to reconstruct just a single segment, leads to huge impact over the computational effort. Finally, despite the efficiency in the algorithm provided by Korobeynikov [3], the approach is developed for the standard SSA algorithm, not benefiting from the other advantages provided by segmentation.

Aiming to contribute in this context, this work details the implementation of a new SSA algorithm, proposed in [6], a companion paper of this Joint Special Issue. This algorithm can be used in many areas of knowledge where the standard SSA is already being used. It is worth to mention as contributions of this new algorithm: improvement in the reconstruction of large time-series; minimization (in some cases mitigation) of the boundary effects associated with segmented analysis; better time-frequency characterization of non-stationary signals [6].

In the sequel, detailed explanations regarding the algorithm and the code are provided. These details include the analysis of the computational effort, revealing that the proposed algorithm is more efficient than standard SSA implementations and very competitive against efficient implementations, as the one proposed by Korobeynikov [3].

2. The proposed methodology

A new algorithm has been proposed in Leles et al. [6], allowing the standard SSA to capture both the varying structures and those that are long-lasting in non-stationary time series, improving the capacity for reconstruction and analysis. This algorithm also enables the use of SSA for large datasets.

The SSA method is more prone to success for short time-series [8] and it is adaptive to the underlaying data [9] so the idea of the proposed algorithm is to take advantage of both these features by using segmentation.

The work [10] shows that in SSA the first component is the one that exhibits the maximum similarity with the spectrum of the original signal, the second one adds more information, and so on, until the last component stores the least amount of information.

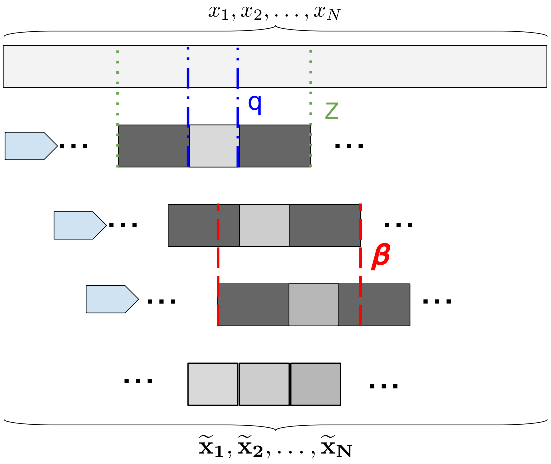


Fig. 1. Segmented SSA.

Ideally, the purpose of segmentation is to divide the original time-series into smaller segments that keep invariant characteristics. If a structure is long-lasting in the time-series, it appears repeatedly over the consecutive segments. Thus, for a stationary time-series, segmentation produces few benefits in comparison with a global analysis. However, if a structure vanishes over time, as occur in non-stationary time-series, it may be captured by a given segment and missed by the remaining ones.

This is the key motivation in the proposed methodology. The original time-series, with length N, is divided into smaller segments, of fixed length Z. For each segment, the SSA algorithm is computed and a reconstructed time-series is obtained. The key innovation in the proposed methodology resides in obtaining the final reconstructed time-series from the segments locally generated. In other words, how to assemble this puzzle together.

3. Methodology description

The idea behind this methodology is better illustrated in Fig. 1. The consecutive segments are not disjointed, they have an amount of overlap, defined as a percentage β. Given a segment, the next one is obtained by displacing the initial value by q samples. Therefore, the overlap percentage is given by $\beta = 100(Z - q)/Z$.

Consider a segment Z. All samples within this segment are used to compute the SSA locally. However, since the SSA method suffers from boundary effects, the extreme points in the left and right edges are discarded. The quantity of discarded samples is given by $\bar{L} = (Z - q)/2$. Only an inner subset of samples q is considered meaningful to represent the local time-series Z. This procedure is illustrated in Fig. 1 for three consecutive segments. The dark areas correspond to reconstructed samples that are discarded. Light gray tones are used to depict the reconstructed samples that are retained. The final reconstruction is given by the concatenation of the inner segments q, which do not overlap. The extreme edges of the original time-series need special attention, which is explained in the next section.

3.1. Algorithm description

The pseudocode of this method is presented in Algorithm 1.

This approach is a modification of the *overlap-save* method, a classic tool to calculate FFT (Fast Fourier Transform) convolution of infinite (and finite) duration time series [11]. This adaptation was necessary because the standard SSA algorithm suffers from

```
**Data**: The original time-series, $(x_1, x_2, \dots, x_N)$.
**Input**: Standard SSA parameters $L$ (the immersion dimension) and $\{I_m\}$ (the grouping size);
    The segmented version parameters $Z$ (local segment length) and $q$ (the amount of samples shifted).
**Output**: Time-series reconstructed by the segmented SSA algorithm, $\widetilde{\mathbf{x}}$.
initialization;
$\bar{L} \leftarrow (Z - q)/2$;
$\mathcal{P} = \lfloor (N - Z)/q \rfloor + 1$;
$s \leftarrow (x_1, x_2, \dots, x_Z)^T$;
$\hat{s} \leftarrow SSA(s, L, \{I_m\})$;
$\widetilde{\mathbf{x}} \leftarrow (\hat{s}_1, \hat{s}_2, \dots, \hat{s}_{\bar{L}+q})^T$;
for $p \leftarrow 2$ to $\mathcal{P} - 1$ do
    $\rho \leftarrow (p - 1)q + 1$;
    $s \leftarrow (x_\rho, x_{\rho+1}, \dots, x_{\rho+Z})^T$;
    $\hat{s} \leftarrow SSA(s, L, \{I_m\})$;
    $\widetilde{\mathbf{x}} \leftarrow (\widetilde{\mathbf{x}}, \hat{s}_{\rho+\bar{L}+1}, \hat{s}_{\rho+\bar{L}+2}, \dots, \hat{s}_{\rho+\bar{L}+q})^T$;
end
$p \leftarrow p + 1$;
$\rho \leftarrow (p - 1)q + 1$;
$s \leftarrow (x_\rho, x_{\rho+1}, \dots, x_{\rho+N})^T$;
$\hat{s} \leftarrow SSA(s, L, \{I_m\})$;
$\widetilde{\mathbf{x}} \leftarrow (\widetilde{\mathbf{x}}, \hat{s}_{\rho+\bar{L}+1}, \hat{s}_{\rho+\bar{L}+2}, \dots, \hat{s}_N)^T$;
```

Algorithm 1: ov-SSA: the segmented version of the standard SSA, a modification proposed in this paper.

Table 1
The input parameters of function `segmentation`.

Parameters	Description
$(x_1, x_2, \dots, x_N)$	The original time-series
L	The embedding dimension
$\{I_m\}$	The set of elementary matrices
Z	Local segment length
q	Amount of samples shifted
Method	SVD calculation: '**bsc**' is the basic version, whilst '**kor**' the efficient one

boundary effects on both sides. In the *overlap-save* method only the initial points must be discarded, because boundary effects occur only at the filtering initialization. Therefore the new methodology is called Overlap SSA (ov-SSA).

3.2. Software description

The software is very simple and consists on some functions written in Matlab. There is a higher level function with syntax

```
y = segmentation(x,L,I,Z,q,method)
```

that executes the segmentation of the original time-series, applies the standard SSA inside the loop shown in Algorithm 1, and, finally, perform the assembling of the consecutive segments, producing the output y: the reconstructed time-series. The input parameters of this function are described in Table 1.

The four steps of standard SSA are implemented as functions. Notice that there are two approaches to perform the SVD step. One is a traditional implementation, whereas the other, provided by `svdecon`,[1] is a more efficient implementation. Such methods are indicated in Table 1 as '**bsc**' and '**kor**', respectively.

3.3. Computational effort

The Windowed SSA (WSSA) algorithm, proposed in Rekapalli and Tiwari [4], can be considered a particular case of the method proposed in Leles et al. [6]. To achieve same results one may just set the overlap parameter to $\beta = 0\%$. This implies no overlap and results in the fastest version of the proposed algorithm. The comparison with Yiou et al. [7] is omitted and the reason is twofold. Firstly, the reconstructed series by Yiou et al. [7] does not have the same length of the original time-series. Secondly, the authors themselves admit that the methodology is more computationally costly than the standard SSA algorithm. To the best of the authors knowledge, these aforementioned works are the ones that deal with segmented versions of SSA algorithm aiming at time-series reconstruction.

3.3.1. Analysis

In this section the computational effort of the proposed method is compared with standard SSA using two implementations. The first case is basic implementation of the SSA, which is denoted by SSA_{bsc}, leading to wort-case complexity given by:

$$\text{SSA}_{\text{bsc}} = O(N^3),$$

where N is the number of samples.

The other implementation, proposed by Korobeynikov [3] and denoted by SSA_{kor}, which is much more efficient, with worst case complexity given by:

$$\text{SSA}_{\text{kor}} = O(\kappa N \log N + \kappa^2 N),$$

where κ is the number of eigenvectors in the grouping stage.

For the proposed approach, there are two possibilities then. If the proposed algorithm is implemented in its basic form one can obtain the following complexity:

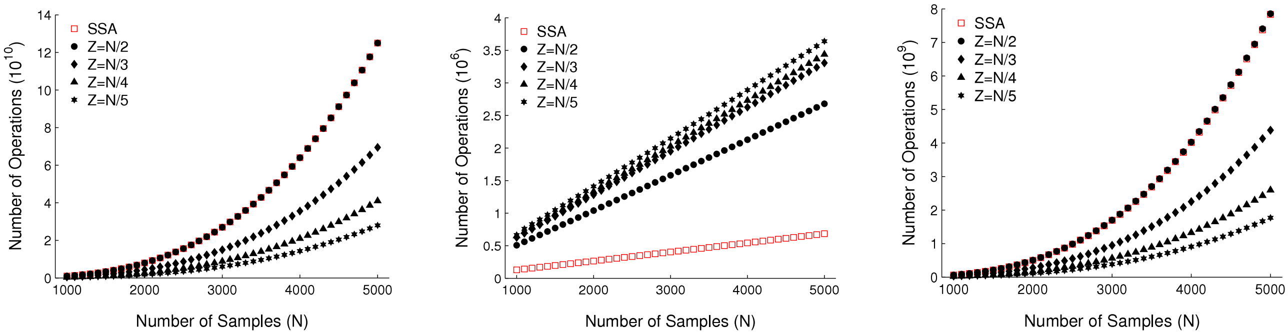
$$\text{ov-SSA}_{\text{bsc}} = O(\mathcal{P} Z^3).$$

However, if the efficient implementation is used

$$\text{ov-SSA}_{\text{kor}} = O(\mathcal{P}[\kappa Z \log Z + \kappa^2 Z]).$$

---

[1] https://www.mathworks.com/matlabcentral/fileexchange/47132-fast-svd-and-pca.



(a) SSA_{bsc} vs ov-SSA_{bsc}. (b) SSA_{kor} vs ov-SSA_{kor}, with $\kappa = 10$. (c) SSA_{kor} ($\kappa = N/4$) vs ov-SSA_{kor} ($\kappa = Z/4$).

Fig. 2. Comparison of computational effort of standard SSA and ov-SSA using two implementations of the SSA algorithm. Overlap percentage $\beta = 85\%$. Different ratios Z/N considered.

Table 2
MAE for the reconstructed time series via ov-SSA for different values of Z and q. The MAE for the SSA was 82.3524. Following [16], the SSA parameters used are: $L = 24$ and $\mathbf{A}_{\{1,2,\ldots,13\}}$.

	$Z = 59$	$Z = 61$	$Z = 63$		$Z = 60$	$Z = 62$	$Z = 64$
$q = 1$	80.5816	78.6206	79.6910	$q = 2$	78.3102	79.3711	79.8009
$q = 3$	80.0970	79.3559	79.7011	$q = 4$	78.1780	80.4561	79.8633

3.3.2. A numeric example

Figure 2 summarizes some relevant results in terms of computational effort. Both kinds of implementations are compared. As the number of total samples (N) is varied, x-axis, the number of operations is computed, y-axis.

The graphics 2(a), consider the basic implementation for standard SSA and the proposed algorithm. Square indicates the results of SSA_{bsc}. The other markers indicate the results produced by ov-SSA_{bsc}, taking distinct lengths Z. By resorting to smaller segments, i.e., by decreasing the ration Z/N, there is a significant reduction in the computational effort. The overlap percentage, given by $\beta = 85\%$.

Graphics shown in 2(b) and 2(c) consider the efficient implementations for the standard SSA and ov-SSA, namely SSA_{kor} and the proposed algorithm, respectively. They are distinguished by the fact that in 2(b) the grouping size is fixed at $\kappa = 10$ for both methods. However, in 2(c) the grouping sizes vary according to the length of the local segment. For SSA_{kor} it was considered $\kappa = N/4$ and for ov-SSA_{kor} it had $\kappa = Z/4$.

Another variable in ov-SSA is the amount of shift between consecutive steps in the prosed algorithm, parameter q. Increasing q diminishes the amount of overlap between consecutive segments. Therefore, with less overlap the number of operations also reduces, improving the results shown. Notice that the maximum value shift must satisfy the constraint $q \leq Z-2L$, to avoid the boundary effects which are discussed in the next section.

4. Illustrative examples

In this section two examples of real-life time-series are presented. In Section 4.1 the proposed method is compared with method of Rekapalli and Tiwari [4] showing the advantages of the proposed approach. On the other hand, in Section 4.2 a time-series used in some SSA studies [12–14] is chosen to demonstrate the capabilities of ov-SSA dealing with short time series.

4.1. Reconstruction of a large dataset

This example illustrates the reconstruction of a large dataset, comprising three-million years mutually consistent records of surface air temperature, sampled at 100 year interval totalizing 30,000 data points. For a discussion about this time-series see [15].

The original time-series has been detrended and is portrayed in Fig. 3(a). In the left column of Fig. 3, three details are shown. The right column displays the same time intervals. The difference is the fact that the left column shows the reconstructed time-series by ov-SSA (blue color) and WSSA (red color) considering $Z = 512$ whereas the right column shows the reconstruction considering $Z = 4096$. In both columns the reconstructed time-series by standard SSA is portrayed as well, in magenta. The dots indicate the beginning or the ending of a given segment, to highlight the concatenation of segments.

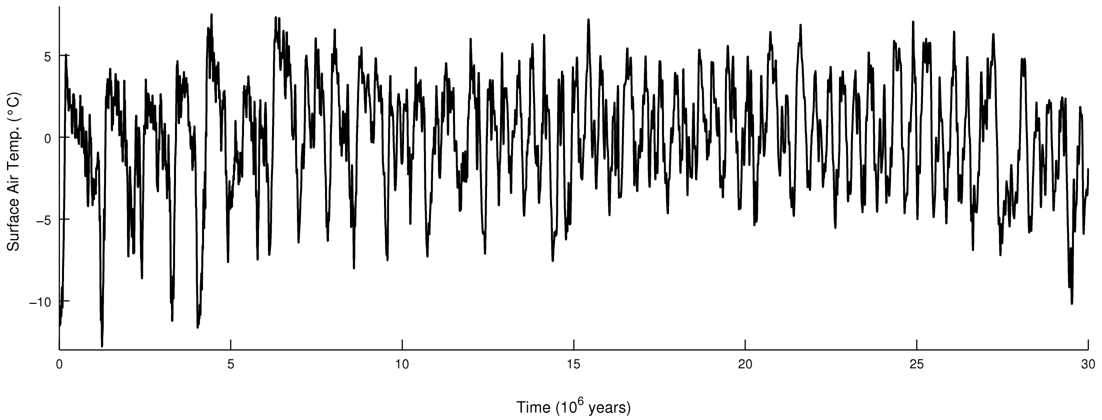
Figs. 3(b) and 3(c) reveal the first issue occurring when the overlap between segments is ignored. The reconstructed time series by WSSA contains abrupt changes not present in the original time-series. This artifact is more pronounced as the segment length is shortened. For the ov-SSA, however the analysis of details 3(f) and 3(g) indicates that the segment length has a negligible effect in discontinuity, being somehow more robust than WSSA.

Another misinformation introduced by disjointed segments can be seen in Fig. 3(d). Although there is not an abrupt change in the extreme samples of the reconstructed time-series it is possible to identify an inflection of the reconstructed time-series that is not present in the original signal.

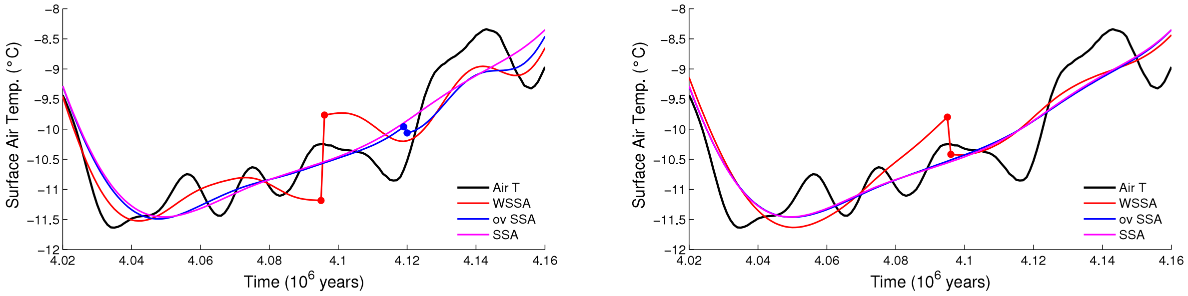
Considering the time-series reconstructed by ov-SSA the discontinuities are minimized, and in some cases completely mitigated, e.g. Figs. 3(f) and 3(g). Also, the transition between consecutive segments is more smooth.

As a final remark, in every detail of Fig. 3 it is possible to check that the reconstructed time-series by the proposed method is among the ones that fit better the original time-series. This aspect is emphasized in a quantitative manner, illustrated by Fig. 4.

The squared error between reconstructed and original time-series was calculated, for several lengths of segment Z. The standard SSA produced the maximum squared error among all methods, as expected. Fig. 4(a) shows the squared error for ov-SSA normalized by the standard SSA error, considering several lengths fro Z. For each length, different percentages of overlap were considered too. These results are compared with the WSSA, which is equivalent to the ov-SSA with $\beta = 0\%$.

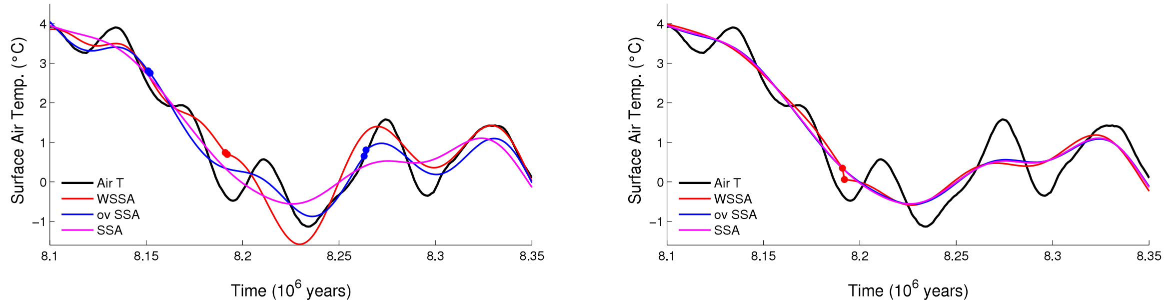


(a) Original time series.



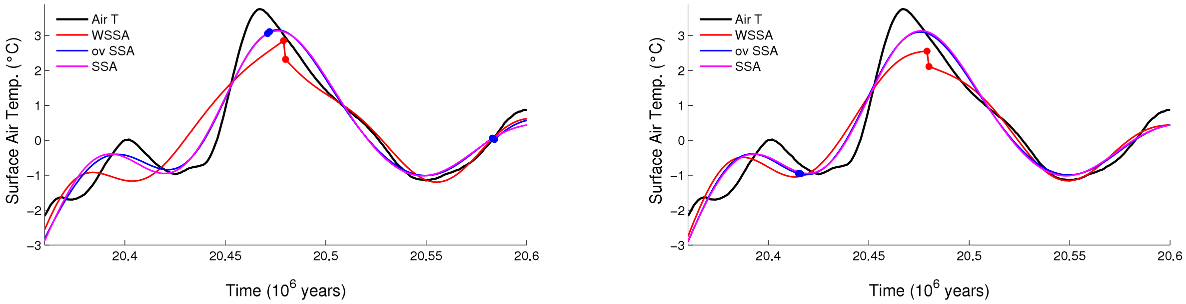
(b) First detail, $Z = 512$.

(c) First detail, $Z = 4096$.



(d) Second detail, $Z = 512$.

(e) Second detail, $Z = 4096$.

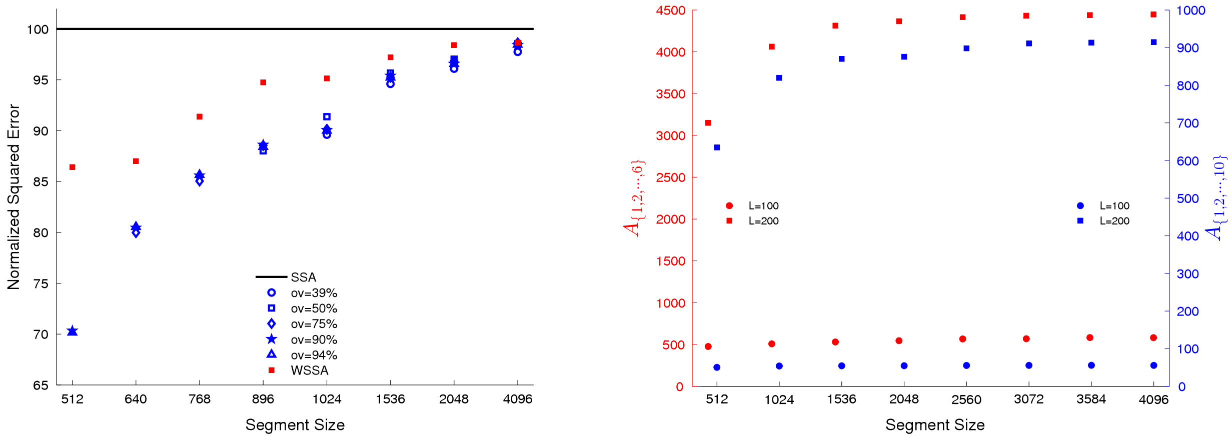


(f) Third detail, $Z = 512$.

(g) Third detail, $Z = 4096$.

Fig. 3. Problems with boundary effect when no overlap is considered (WSSA) and the reduction/mitigation of this issue in the proposed approach (OV-SSA). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

This numerical analysis reveals that the proposed algorithm can improve the time-series reconstruction with respect to the standard SSA and WSSA. In some cases the decrease in the squared error was almost 30% and 20%, when compared to the standard SSA and WSSA, respectively. As it is expected, as the length of the segment increases the improvements are less noticeable and asymptotically converges to the standard SSA, which is the ultimate case consisting of a single segment comprising the whole time-series.



(a) Normalized error with distinct Z.

(b) Squared error for the ov-SSA with different parameters.

Fig. 4. Quantitative metrics concerning the reliability of the time-series reconstruction for comparison of algorithms. Influence of the overlap in the reliability of the approximation (a). Analysis of immersion dimension and size of grouping (b).

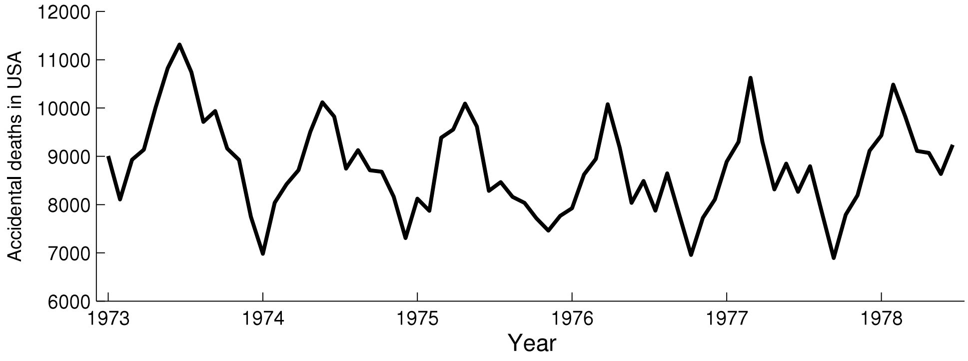


Fig. 5. Monthly accidental deaths in the USA between 1973 and 1978.

Table 3
MAE for the reconstructed time series via ov-SSA for different values of Z and q. The MAE for the SSA was 164.1134. Following [13], the SSA parameters used are: $L = 24$ and $\mathbf{A}_{\{1,2,3,5,6,7,8,10,14\}}$.

	$Z = 59$	$Z = 61$	$Z = 63$		$Z = 60$	$Z = 62$	$Z = 64$
$q = 1$	157.6796	158.5879	159.1798	$q = 2$	156.6042	158.1741	162.9879
$q = 3$	158.2749	161.8584	160.0026	$q = 4$	156.9237	159.5746	162.5992

4.2. Reconstruction of a short dataset

To discuss other feature of ov-SSA, the Monthly accidental deaths[2] in the USA, between 1973 and 1978, is used. This time series, depicted in Fig. 5, was used in different SSA studies [13,14,16]. Hassani [16] applied the SSA technique to this time-series illustrating some of SSA details and several heuristics to SSA parameter selection to extract trend, oscillation and noise components of a time-series.

It worth to highlight that this time-series consists of only 72 samples and the minimum size of the segment Z can be equal to 50. Nonetheless some differences can be noted by the segmented approach.

Tables 2 and 3 present the MAE comparison for ov-SSA using different values of Z and q against the SSA for the parameters defined in Hassani [16] and Hassani et al. [13], respectively.

As can be seen by analysis of Tables 2 and 3, ov-SSA was able to reconstruct this (short) time-series providing, for different combinations of Z and q better results than SSA. It should be highlighted that those SSA parameters were selected for basic SSA.

5. Impact

Large datasets arise for many applications and different fields: economics and finance, bioinformatics, telecommunications, web mining. The capacity to extract relevant information from large datasets is therefore of uttermost importance. In this context SSA is a powerful tool for analysis that has gaining momentum. This software can help towards the usage of SSA in existing and new applications where otherwise there were difficulties associated with the size of the dataset. Furthermore, the segmented approach is useful even for short-time series, since provides an alternative to capture time-varying features present in the observed time-series.

The software can help the development of the SSA methodology itself. Since there is a growing interest in the mining of datasets,

---

[2] This time-series is freely available for download at: https://datamarket.com/data/set/22p0/accidental-deaths-in-usa-monthly-1973-1978.

arises the interest in a high level form to representation of meaningful information. The usage of a segmented version of the SSA algorithm can be beneficial to extract local information, which could be hidden in a standard approach. For a further discussion, see [6]. Also, the SSA community now has a new tool available to support investigations concerning the optimal choice of segment length.

The MATLAB architecture was chosen because is a widespread language across many fields of science and engineering. Therefore many contents related to SSA has been and may be developed, allowing users to easily and seemingly adapt the proposed software to their specific needs.

This algorithm does not intend to provide an improvement of the standard SSA for every applications. However, the user has a flexible tool and can adjust the overlap and segment size parameters to progressively analyze the data from a local, or segmented, manner until a global manner. If $Z = L$ the standard SSA approach is recovered.

For further discussion about the possible impacts and recommendations for usage the reader is referred to [6].

6. Conclusions

In this paper a implementation of a new algorithm, the Overlapped SSA (ov-SSA), has been presented. This new method can improve the time-series reconstruction and discrimination of dominant structures with respect to the standard SSA. It also contributes to the usage of SSA for large datasets.

Compared to existing algorithms the ov-SSA promote some improvements: the reduction of boundary effects; the segment concatenation is more smooth, once abrupt changes are decreased/mitigated; the reconstructed time-series has the same length of the original one; the local analysis has little impact over the computational effort, in comparison with efficient implementations.

Acknowledgments

We acknowledge the Brazilian agencies CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior), CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico), and FAPEMIG (Fundação de Amparo à Pesquisa do Estado de Minas Gerais) for partial support of this work.

References

[1] Golyandina N, Zhigljavsky A. Singular spectrum analysis for time series. Springer Briefs in Statistics; 2013.
[2] Leles MCR, Mozelli LA, Guimarães HN. New trend-following indicator: Using SSA to design trading rules. Fluct Noise Lett 2017;16:1750016.
[3] Korobeynikov A. Computation- and space-efficient implementation of SSA. Stat Interface 2010;3:357–68.
[4] Rekapalli R, Tiwari RK. Windowed SSA (singular spectral analysis) for geophysical time series analysis. J Geol Resour Eng 2014;3:167–73.
[5] Azarbad M, Azami H, Sanei S, Ebrahimzadeh A. A Time-Frequency approach for EEG signal segmentation. J AI Data Min 2014;2(1):63–71.
[6] Leles MCR, Guimaraes HN, Sansao JP, Mozelli LA. Improving reconstruction of time-series with singular spectrum analysis: A segmentation approach. Digit Signal Process 2016;1. https://doi.org/10.1016/j.dsp.2017.10.025, in press.
[7] Yiou P, Sornette D, Ghil M. Data-adaptive wavelets and multi-scale singular-spectrum analysis. Physica D 2000;142(3):254–90.
[8] Vautard R, Yiou P, Ghil M. Singular-spectrum analysis: A toolkit for short, noisy chaotic signals. Physica D 1992;58:95–126.
[9] Jemwa GT, Aldrich C. Classification of process dynamics with monte carlo singular spectrum analysis. Comput Chem Eng 2006;30(5):816–31.
[10] Kume K. Interpretation of singular Spectrum Analysis as complete eigenfilter decomposition. Adv Adapt Data Anal 2012;04(04):1250023.
[11] Brigham E. Fast Fourier transform and its applications. Prentice Hall; 1988.
[12] Hassani H, Heravi S, Zhigljavsky A. Forecasting european industrial production with singular Spectrum Analysis. Int J Forecast 2009;25:103–18.
[13] Hassani H, Ghodsi Z, Silva ES, Heravi S. From nature to maths: Improving forecasting performance in subspace-based methods using genetics Colonial Theory. Digit Signal Process 2016;51:101–9.
[14] Wang R, Ma H-G, Qin J-Q, Feng X-W, Liu H-M. Analysis of death series by ssa based bss technique. In: Information, communications and signal processing, 2015 10th international conference on. IEEE; 2015. p. 1–5.
[15] Bintanja R, van de Wal RSW. North american ice-sheet dynamics and the onset of 100,000-year glacial cycles. Nature 2008;454:869–72.
[16] Hassani H. Singular spectrum analysis: Methodology and comparison. J Data Sci 2007;5(2):259–67.