Time Series Segmentation of Paleoclimate Tipping Points by an Evolutionary Algorithm*

M. Pérez-Ortiz[1], P.A. Gutiérrez[1], J. Sánchez-Monedero[1], C. Hervás-Martínez[1], Athanasia Nikolaou[2], Isabelle Dicaire[2], and Francisco Fernández-Navarro[2]

[1] University of Córdoba, Dept. of Computer Science and Numerical Analysis, Rabanales Campus, Albert Einstein building, 14071 - Córdoba, Spain
{i82perom,pagutierrez,jsanchezm,chervas}@uco.es
[2] Advanced Concepts Team, European Space Research and Technology Centre (ESTEC), European Space Agency (ESA), Noordwijk, Netherlands
{athanasia.nikolaou,isabelle.dicaire,francisco.fernandez.navarro}@esa.int

Abstract. Recent studies propose that some dynamical systems, such as climate, ecological and financial systems, among others, present critical transition points named to as *tipping points* (TP). Climate TPs can severely affect millions of lives on Earth so that an active scientific community is working on finding early warning signals. This paper deals with the segmentation of a paleoclimate time series to find segments sharing common patterns with the purpose of finding one or more kinds of segments corresponding to TPs. Due to the limitations of classical statistical methods, we propose the use of a genetic algorithm to automatically segment the series together with a method to perform time series segmentation comparisons. Without a priori information, the method clusters together most of the TPs and avoids false positives, which is a promising result given the challenging nature of the problem.

Keywords: Time series segmentation, genetic algorithms, clustering, paleoclimate data, tipping points, abrupt climate change.

1 Introduction

In contrast to the famous statement of Linnaeus (1751) "*natura non facit saltus*" (or nature makes no leaps), it has been proven that some points of no return, thresholds and phase changes are widespread in nature and these are often non linear [1]. Such events can be rarely anticipated and some of them can have detrimental consequences on Earth's climate and large-scale impacts on human and ecological systems. Therefore, this increases the imperious necessity of studying, analysing and developing techniques for characterizing them in order to

* This work has been subsidized by the Ariadna project 13-9202 of the European Space Agency. The research work of M. Pérez-Ortiz, P.A. Gutiérrez, J. Sánchez-Monedero and C. Hervás-Martínez is partially funded by the TIN2011-22794 project of the Spanish Ministerial Commission of Science and Technology (MICYT), FEDER funds and the P11-TIC-7508 project of the "Junta de Andalucía" (Spain).

M. Polycarpou et al. (Eds.): HAIS 2014, LNAI 8480, pp. 318–329, 2014.

construct reliable early warning systems. Although the human being have influenced their local environment for millennia, e.g. reducing biodiversity, it is now, since the industrial revolution, that truly global changes are been noticed [2]. Examples that are currently receiving attention include the potential collapse of the Atlantic thermohaline circulation, the dieback of the Amazon rainforest or the decay of the Greenland ice sheet [1]. Formally, a climate "tipping point" (also known as "little things can make a big difference") occurs when a small change in forcing triggers a strongly nonlinear response in the internal dynamics of part of the climate system, qualitatively changing its future state.

The critical relevance of early TPs detection has produced a growing attention of the scientific community. Lenton differences between several types of TPs, and presents some indicators that can help to detect them, such as the increase of autocorrelation of the series values [3]. In [4], more concrete techniques regarding data processing and indicators are presented. They study a bank of methods using only simulated ecological data concluding, in concordance with the literature, that there is no unique best indicator for identifying an upcoming transition. They also conclude that all the methods require specific data-treatment. Up to our knowledge, all previous works tackle the TPs detection with statistical methods trying to select (by trial and error) the method more suitable to detect those transitions. They require an intensive data preprocessing that include, for instance, the use of Gaussian filters or rolling windows that introduces extra parameters (such as the width of the Gaussian function or size of the window) that need to be optimised [3,4]. The main limitation behind these methods is that different TPs and different statistical descriptors require different and specific treatments.

This paper deals with climate time series segmentation. We introduce a segmentation method as a first step to better understand the time series. This segmentation provides a more compact representation of the time series through splitting it into segments with similar behaviour [5]. A segmentation analysis avoids the necessity of specifying predefined sliding windows for the different TPs, which is one of the main difficulties of previous TP detection methods [4]. Moreover, the segmentation algorithm is able to detect differences between the TPs. We address the segmentation problem as a heuristic search problem with the proposal of a Genetic Algorithm (GA) to overcome the limitations of traditional statistical methods. The GA segments the data trying to obtain diverse clusters of segments based on six statistical properties. Measuring the quality of a segmentation can be only achieved by expert evaluation of the solutions given by the algorithm. An important contribution of this paper is a quantitative method to perform comparisons with respect to an expected ideal segmentation of the series to assess the robustness and stability of the method. This method allows evaluating a segmentation algorithm with a minimal effort by the expert, who has only to provide the ideal segmentation. We test the proposal with data collected within the North Greenland Ice Core Project δ^{18}O ice core data [6,15] which includes climate records from -60,000 years to the present.

The rest of the paper is organised as follows. Section 2 presents the segmentation algorithm, while Section 3 presents a proposal for segmentation comparison and discusses the experimental results. The last section depicts the conclusions.

2 Segmentation Algorithm

Given a time series $Y = \{y_n\}_{n=1}^{N}$, our objective is to divide the values of y_n into m consecutive subsets or segments. These segments should present a homogeneous behaviour regarding the values of y_n. This is done by partitioning the time indexes ($n = 1, \ldots, N$) into segments: $s_1 = \{y_1, \ldots, y_{t_1}\}, s_2 = \{y_{t_1}, \ldots, y_{t_2}\}, \ldots, s_m = \{y_{t_{m-1}}, \ldots, y_N\}$, where t's are the cut points and are subscripted in ascending order ($t_1 < t_2 < t_{m-1}$). The cut points belong to two segments (the one before and the one after, which allows to analyse consistently the transition from one segment to the next). The integer m and the cut points $t_i, i = 1, \ldots, m-1$, are the parameters to be determined by the algorithm. As done in [7], we extend this setting by trying to group the segments into k different classes or clusters ($k < m$), where k is a parameter defined by the user. In this way, each s_l segment will be associated to a class label: (s_1, z_1), ... ,(s_m, z_m), where z_l, $l = 1, \ldots, m$, is the class label of the l-th segment and takes values in a set of k different labels, $z_l \in \{\mathcal{C}_1, \ldots, \mathcal{C}_k\}$.

2.1 Summary of the Algorithm

The Genetic Algorithm considered in this paper can be included in the area of time series segmentation [5,8,9,10]. Each possible segmentation is represented as an array of binary values (chromosome representation), where a value of 1 represents a cut point. The evolution starts from a population of randomly generated segmentations. Mutation and crossover operators are applied to explore and exploit the search space. This procedure is repeated g generations. To evaluate a solution (or segmentation), we select a set of statistics to be calculated for each segment. Then, similar segments are grouped using a clustering process. The different characteristics of the GA are defined in the following subsections.

2.2 Chromosome Representation

As stated before, each individual chromosome consists of an array of binary values, where the length of the chromosome is the time series length, N. Each position c_i stores whether the time index t_i of the time series represents a cut point for the evaluated solution[1]. In this sense, for a given segment s_i delimited by the cut points t_{i-1} and t_i ($t_{i-1} < t_i$), the corresponding chromosome values will be $c_{i-1} = 1$, $c_i = 1$ and $c_l = 0, \forall l | t_{i-1} < l < t_i$.

[1] Note that the first and last points of the chromosome are considered as cut points.

2.3 Initial Population

The population of the GA is a set of binary vectors of length N. In order to initialise the population, an average segment length has to be specified by the user ($\overline{sl}$). Taking into account that the cut points belong to two segments, the number of cut points will be $\overline{m} = \lceil \frac{N}{\overline{sl}-1} \rceil$, so the chromosomes are binary arrays where $\overline{m}$ random positions are 1s and the rest are 0s.

2.4 Fitness Evaluation

Evaluation of the quality of a segmentation consists of three different steps: extracting the characteristics of the segments, applying a clustering process and measuring the quality of this clustering.

Extracting Segment Characteristics. Given that the segments in a chromosome can have different length, an approach is designed to project all the segments into the same dimensional space. Six statistical metrics are considered and measured for all chromosome segments. Then, the similarities between segments can be calculated in the 6-dimensional space. Consider s_s as a segment fulfilling the previously stated conditions (i.e., s_s is a segment delimited by the cut points t_{s-1} and t_s, where the segment length is $t_s - t_{s-1} + 1$). The mapping is done by the function $f : \mathbb{R}^{(t_s - t_{s-1}+1)} \rightarrow \mathbb{R}^6$, in the following way:

$$f(s_s) = \left(S_s^2, \gamma_{1s}, \gamma_{2s}, a_s, MSE_s, AC_s\right) \tag{1}$$

where the different characteristics are defined as:

1. Variance (S_s^2): It measures the variability of the segment:

$$S_s^2 = \frac{1}{t_s - t_{s-1}+1} \sum_{i=t_{s-1}}^{t_s} \left(y_i - \overline{y_s}\right)^2, \tag{2}$$

   where y_i are the time series values of the segment, and $\overline{y_s}$ is the average value of the segment.
2. Skewness (γ_{1s}): It represents the (vertical) asymmetry of the distribution of the series values in the segment with respect to the arithmetic mean:

$$\gamma_{1s} = \frac{\frac{1}{t_s - t_{s-1}+1} \sum_{i=t_{s-1}}^{t_s} (y_i - \overline{y_s})^3}{S_s^3}, \tag{3}$$

   where S_s is the standard deviation of the s-th segment.
3. Kurtosis (γ_{2s}): This statistic is related to the degree of concentration that the values present around the mean of the distribution:

$$\gamma_{2s} = \frac{\frac{1}{t_s - t_{s-1}+1} \sum_{i=t_{s-1}}^{t_s} (y_i - \overline{y_s})^4}{S_s^4} - 3. \tag{4}$$

4. Slope of a linear regression over the points of the segment (a_s): A linear model is constructed for every segment trying to achieve the best linear

approximation of the points of the time series in the evaluated segment. The slope is a measure of the general tendency of the segment:

$$a_s = \frac{S_s^{yt}}{(S_s^t)^2}, \tag{5}$$

where, for the s-th segment, S_s^{yt} is the covariance between the time indexes, t, and the time series values, y; and S_s^t is the standard deviation of the time values. Covariance S_s^{yt} is defined by:

$$S_s^{yt} = \frac{1}{t_s - t_{s-1} + 1} \sum_{i=t_{s-1}}^{t_s} (i - \overline{t_s}) \cdot (y_i - \overline{y_s}). \tag{6}$$

5. Mean Squared Error (MSE_s): Considering the same linear model than the one used for the slope, we measure the error (MSE_s) of this linear fitting:

$$MSE_s = S_s^2 \cdot (1 - r_s^2), \text{ where } r_s^2 = \frac{S_s^{yt}}{S_s^2 \cdot (S_s^t)^2}. \tag{7}$$

6. Autocorrelation coefficient (AC_s): This a measure of the correlation between the current values of the time series and the previous ones:

$$AC_s = \frac{\sum_{i=t_{s-1}}^{t_s} (y_i - \overline{y_s}) \cdot (y_{i+1} - \overline{y_s})}{S_s^2}. \tag{8}$$

Clustering Process: *k*-means. A clustering process is applied to group similar segments (taking into account the six selected statistical measures). For simplicity, the algorithm chosen for the clustering step is the well-known k-means. Before the clustering algorithm, note that a normalisation of the values of the segment metrics is conducted, as the distance from each segment to its centroid strongly depends on the range of values of each metric (e.g. variance can have a much broader range of variation than skewness).

In the classic k-means, the initial centroids are randomly chosen from the set of patterns. Instead, we have developed a deterministic process to select these centroids which ensures that a chromosome will always present the same fitness. First, we choose the feature with the maximum standard deviation. The first initial centroid will be the segment with the highest value for this feature. The second one will be the segment with the highest Euclidean distance from the first centroid. The third centroid will be that which is farthest from both, and so on. This assures a deterministic initialisation, at the same time that the initial centroids are as far as possible from each other, favouring centroids diversity.

Measuring the Quality of the Clustering Process. The last step of the evaluation of the chromosome is to measure how well the segments are grouped (compactness of the clustering). It is clear that different clustering algorithms usually lead to different clusters or reveal different clustering structures. In this sense, the problem of objectively and quantitatively evaluating the clustering results is particularly important and this is known in the literature as cluster validation. There are two different testing criteria for this purpose [11]:

external criteria and internal criteria. When a clustering result is evaluated based on the data that was clustered itself, this is called internal evaluation. In external evaluation, clustering results are evaluated using for example known class labels. Based on these concepts, the internal criteria evaluation metrics will be a suitable option for the evolution, because the GA is not given any a priori information of the segments to be found. Note that the segments metrics are normalised at this step as well. We have considered four different metrics:

1. *Sum of squared errors (SSE)*: The simplest error measure is the sum of squared errors (considering errors as the distance from each point to their centroid), i.e.:
$$SSE = \frac{1}{N} \sum_{i=1}^{k} \sum_{\mathbf{x} \in \mathcal{C}_i} d(\mathbf{x}, \mathbf{c}_i)^2, \tag{9}$$
where k is the number of clusters, $\mathbf{c}_i$ is the centroid of cluster $\mathcal{C}_i$ and $d(\mathbf{x}, \mathbf{c}_i)$ is the Euclidean distance between pattern $\mathbf{x}$ and centroid $\mathbf{c}_i$. This function does not prevent clusters to fall very close in the clustering space. As this index has to be minimised, the fitness will be defined as $f = \frac{1}{1+SSE}$.
2. *Caliński and Harabasz index (CH)*: This index has been found to be one of the best performing ones for adjusting the value of k. It is defined as:
$$CH = \frac{\mathtt{Tr}(\mathbf{S}_B) \cdot (N-k)}{\mathtt{Tr}(\mathbf{S}_W) \cdot (k-1)}, \tag{10}$$
where N is the number of patterns, and $\mathtt{Tr}(\mathbf{S}_B)$ and $\mathtt{Tr}(\mathbf{S}_W)$ are the trace of the between and within-class scatter matrix, respectively. Note that the value of k will be fixed in our algorithm. As this index has to be maximised, the fitness will be defined as $f = CH$.
3. *Davies-Bouldin index (DB)*: This index also attempts to maximize the between-cluster distance while minimising the distance between the cluster centroids to the rest of points. It is calculated as follows:
$$DB = \frac{1}{k} \sum_{i=1}^{k} \max_{i \neq j} \frac{\alpha_i + \alpha_j}{d(\mathbf{c}_i, \mathbf{c}_j)}, \tag{11}$$
where α_i is the average distance of all elements in cluster $\mathcal{C}_i$ to centroid $\mathbf{c}_i$, and $d(\mathbf{c}_i, \mathbf{c}_j)$ is the distance between centroids $\mathbf{c}_i$ and $\mathbf{c}_j$. As this index has to be minimised, the fitness will be defined as $f = \frac{1}{1+DB}$.
4. *Dunn index (DU)*: The Dunn index attempts to identify clusters that are compact and and well-separated. In this case, the distance between two clusters is defined as $d(\mathcal{C}_i, \mathcal{C}_j) = \min_{\mathbf{x} \in \mathcal{C}_i, \mathbf{y} \in \mathcal{C}_j} d(\mathbf{x}, \mathbf{y})$, that is, the minimum distance between a pair of points $\mathbf{x}$ and $\mathbf{y}$ belonging to $\mathcal{C}_i$ and $\mathcal{C}_j$. Furthermore, we could define the diameter $\mathtt{diam}(\mathcal{C}_i)$ of cluster $\mathcal{C}_i$ as the maximum distance between two of its members, such as: $\mathtt{diam}(\mathcal{C}_i) = \max_{\mathbf{x}, \mathbf{y} \in \mathcal{C}_i} d(\mathbf{x}, \mathbf{y})$. Then, the Dunn index is constructed as:
$$DU = \min_{i=1,\ldots,k} \left(\min_{j=i+1,\ldots,k} \left(\frac{d(\mathcal{C}_i, \mathcal{C}_j)}{\max_{l=1,\ldots,k} \mathtt{diam}(\mathcal{C}_l)} \right) \right). \tag{12}$$
The Dunn index has been found to be very sensitive to noise, but this disadvantage can be avoided by considering different definitions of cluster distance

or cluster diameter. For example, as suggested in [11], the cluster diameter can be computed as:

$$\texttt{diam}(\mathcal{C}_i) = \frac{1}{N_{\mathcal{C}_i}(N_{\mathcal{C}_i}-1)} \sum_{\mathbf{x},\mathbf{y}\in\mathcal{C}_i} d(\mathbf{x},\mathbf{y}), \tag{13}$$

where $N_{\mathcal{C}_i}$ is the number of patterns belonging to cluster $\mathcal{C}_i$. This cluster diameter estimation has been found to be more robust in the presence of noise. As this index has to be maximised, the fitness will be $f = DU$.

2.5 Selection and Replacement Processes

All individuals will be considered for reproduction and generation of offspring, promoting a greater diversity because all individuals are possible parents. After the application of the genetic operators, the offspring and the parent population are joined and a replacement process is performed by roulette wheel selection. The selection probability for each individual chromosome is calculated from its fitness value. The roulette wheel process is repeated as many times as the population size minus one, and the last place is kept for the best segmentation of the previous generation, thus being an elitist algorithm. As can be seen, the selection process promotes diversity, while the replacement process promotes elitism.

2.6 Mutation Operator

Two mutation operators are included in the GA with the aim of reducing the dependency with respect to the initial population and escaping from local optima. The probability p_{m} of performing any mutation is decided by the user. Once a mutation is decided to be performed, the kind of perturbation applied to the chromosome is randomly selected from the following two: 1) add or remove (with the same probability) a given number of cut points of the segmentation; and 2) move a given number of cut points of the segmentation towards the left or the right (with the same probability).

For all the mutations, the number of cut points to be mutated is decided by a user parameter as a percentage of the current number of cut points. When moving cut points to the right or the left, each selected cut point is randomly pushed towards the previous or the following cut point (with the constraint that it never reaches the previous or the next point).

2.7 Crossover Operator

The algorithm includes a crossover operator, whose main function is to perform an exploitation of the existing solutions. For each parent individual, the crossover operator is applied with a given probability p_{c}. The operator randomly selects the other parent and a time index. It interchanges the left and right parts of the chromosomes selected with respect to the time index.

3 Experiments

The dataset chosen for this study is the North Greenland Ice Core Project (NGRIP) δ^{18}O ice core data [6,15]. The δ^{18}O water isotope record is used as a proxy for past atmospheric temperature. We focus on the 20-yr resolution δ^{18}O isotope records. The dataset is pre-processed by obtaining a 5-point average in order to reduce short-term fluctuations within the data. In this way, the time series we have considered is $\{y_n^*\}_{n=1}^{N/5}$ with $y_i^* = \frac{1}{5}\sum_{j=5i}^{5i+4} y_i$.

3.1 Experimental Setting

The experimental design is presented in this subsection. The GA was configured with the following parameters: the number of individuals of the population is $P = 100$. The crossover probability is $p_c = 0.8$ and the mutation probability $p_m = 0.2$. The percentage of cut points to be mutated is the integer part of the 20% of the number of cut points, and the average segment length for the initialisation is $\overline{sl} = 4$. The maximum number of generations is set to $g = 100$, and the k-means clustering process is allowed a maximum of 20 iterations. These parameters were optimised by a trial and error procedure, although the algorithm showed a very robust performance to their values. The most important parameters for the final performance of the algorithm were $\overline{sl}$ and k.

We performed different experiments considering the 4 different fitness functions presented in Section 2.4 and different values of k for the k-means algorithm ($k = 2, \ldots, 6$). It is important to recall that the algorithm estimates the optimal segments and clusters them without any prior information of the DO events. The only information given to the algorithm is the time series and the statistic characteristics to use for the clustering in order to validate whether the statistics proposed in the literature are useful for characterising paleoclimate TPs in general. Given the stochastic nature of GAs, the algorithm was run 30 times with different seeds to evaluate its stability and robustness.

3.2 Evaluation Metrics

In order to evaluate the results of the algorithm, two evaluation metrics were used. These measures analyse both the homogeneity of cluster assignation with respect to the DO events and the robustness of the results obtained from different seeds. They are not included in the fitness function, serving only as an automatic way of evaluating the quality of the segmentation, avoiding the intervention of the expert. Both are indexes comparing two different clustering partitions:

1. *Rand index (RI)* This metric is particularly useful for data clustering evaluation [12]. It is related to the accuracy, but is applicable even when class labels are not available for the data, as in our case. A set $Y = \{y_n\}_{n=1}^{N}$ is given (in our case, the time series), and two clustering partitions of Y are to be compared: $X = \{X_1, \ldots, X_r\}$ and $Z = \{Z_1, \ldots, Z_s\}$. For a given segmentation, the partitions are defined in the following way: X_l is a set containing

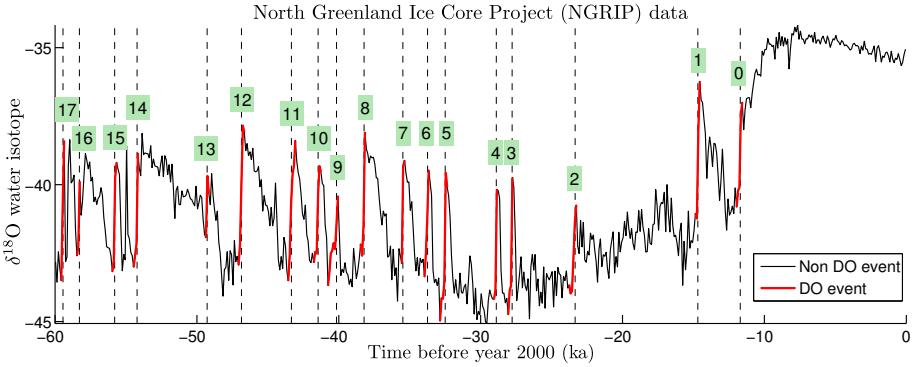

Fig. 1. Representation of the ideal segmentation and the different DO events

every $y_i \in s_s, s_s \in \mathcal{C}_l$, i.e. the partitions are based on the label assigned to each time series value y_i from the current segmentation. The following two numbers are defined: a (number of pairs in Y that are in the same set in X and Z) and b (number of pairs in Y that are in different sets in X and Z). Then, the Rand index is defined as: $RI = (a+b)/\binom{n}{2}$. This metric has a value between 0 and 1, with 0 indicating that the two partitions do not agree on any pair of points and 1 indicating that they are exactly the same.

2. *Adjusted rand index (ARI)*: It is a corrected version of the RI [13] trying to fix some known problems with the RI, e.g. the expected value of the RI of two random partitions does not take a constant value and it approaches its upper limit of unity as the number of clusters increases. ARI values range from -1 to $+1$, yielding negative values if the index is less than the expected index. The detailed formulation can be found in [13].

In order to evaluate the segmentation returned by the algorithm, we compare it with an ideal segmentation[2]. The ideal segmentation (Fig. 1) has been designed by examining the literature about Dansgaard-Oeschger (DO) events, which are associated to TPs. In the Figure, the onsets of the DO events (in a first approximation, we do not consider the error margin) reported in [15] are represented by vertical lines and the segments covering the period precursor to the DO events (which we hypothesize as TP) are delimited by the slope close to the corresponding onset. The closer the segmentation returned by the GA is to this ideal segmentation, the better the segmentation. To perform this comparison, RI and ARI indexes will be used (ARI_Ideal and RI_Ideal).

Given that the wishful ideal segmentation would be binary (non DO event or DO event) and the segmentation returned by the GA can have a value of $k > 2$, we need to binarise the segmentation of the GA (i.e. decide which clusters

[2] Hypothetically ideal segmentation, based on the available data. The hypothesis is that the onset of the DO events is detected from combined analysis of benthic sediment data and ice core analysis [14]. Those data do not always agree, therefore part of the error margin. The method of timing contributes the rest of the error.

Table 1. NGRIP average segmentation results for different algorithm settings

Fitness	k	ARI_Ideal	RI_Ideal	ARI_Seeds	RI_Seeds
DB	5	$\mathbf{0.315 \pm 0.060}$	$\mathit{0.777 \pm 0.015}$	$\mathbf{0.346 \pm 0.078}$	$\mathbf{0.727 \pm 0.040}$
DU	5	$\mathit{0.308 \pm 0.067}$	$\mathbf{0.788 \pm 0.018}$	$\mathit{0.341 \pm 0.092}$	$\mathbf{0.727 \pm 0.046}$
CH	5	0.260 ± 0.073	0.772 ± 0.008	0.223 ± 0.105	0.644 ± 0.074
SSE	5	0.279 ± 0.048	0.770 ± 0.018	0.057 ± 0.018	0.638 ± 0.017
Fitness	k	ARI_TPs	RI_TPs	ARI_Seeds	RI_Seeds
DB	2	0.171 ± 0.132	0.766 ± 0.001	0.258 ± 0.292	$\mathbf{0.821 \pm 0.081}$
DB	3	0.257 ± 0.081	0.758 ± 0.013	$\mathit{0.411 \pm 0.152}$	$\mathit{0.780 \pm 0.046}$
DB	4	$\mathit{0.304 \pm 0.045}$	0.773 ± 0.009	$\mathbf{0.412 \pm 0.080}$	0.761 ± 0.037
DB	5	$\mathbf{0.315 \pm 0.060}$	$\mathit{0.777 \pm 0.015}$	0.346 ± 0.078	0.727 ± 0.040
DB	6	0.286 ± 0.075	$\mathbf{0.779 \pm 0.014}$	0.214 ± 0.109	0.615 ± 0.084

represent the DO events and which not). Preliminary experiments revealed that DO events were usually grouped under one or two clusters, so we evaluated ARI_Ideal and RI_Ideal for all possible combinations of one or two clusters. The final value was the maximum ARI_Ideal and RI_Ideal values of all these combinations. Moreover, the stability of the GA was estimated by comparing the 30 segmentations from the different runs. This was done by averaging RI and ARI comparing all possible pairs of segmentations (ARI_Seeds and RI_Seeds).

3.3 Results

All these results are included in Table 1. The first part of the table compares the different fitness functions for a predefined value of $k = 5$ (as we initially observed that this was obtaining suitable results). As can be seen, both DB and DU fitness functions obtain very good segmentation quality and stability, although DB performs slightly better. In contrast, CH and SSE are performing poorly in both scenarios (it is noteworthy the very low stability obtained by the SSE fitness function, which may be due to the fact that it only minimises the intra-cluster distances and obviates the inter-cluster distances). The result that the algorithm is robust and stable to different initialisations is crucial for the following parts of the study (i.e. develop an early warning system for TPs of climatic component). Concerning the experiment that studies different values of k, it can be seen that $k = 5$ is indeed the optimal value for the segmentation. This result indicates that the concept and nature of DO events is too complex to only consider a binary approach (TPs versus non TPs). The climate system exhibits a dynamical behaviour with intrinsic variability hence a binary approach is not able to encompass all features present within a DO event, being $k = 5$ a reasonable choice. Moreover, the method can group several DO events together and is still a useful tool to better understand the behaviour of DO events.

The segmentation obtaining the highest ARI_Ideal metric (with a value of 0.498) for the fitness function DB, along with a representation of the 18 DO events can be seen in Fig. 2. The segments have been coloured according to their cluster assignation. The clusters associated to the DO events are $\mathcal{C}_1$ and $\mathcal{C}_5$. If we compare this segmentation to the one in Fig. 1, we can see that almost all DO

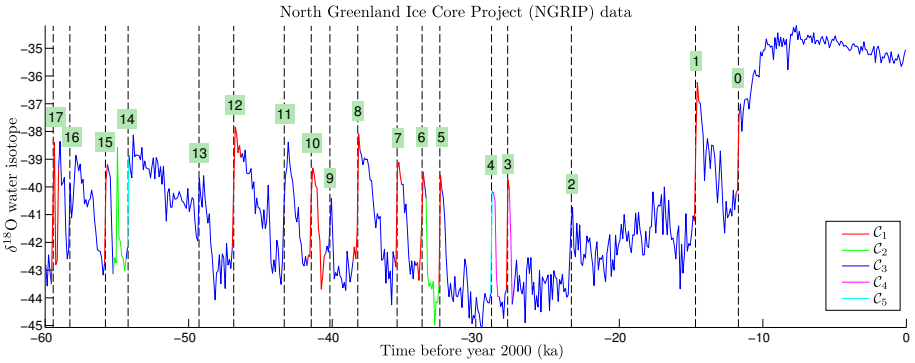

Fig. 2. Best time series cluster assignation after the evolutionary process

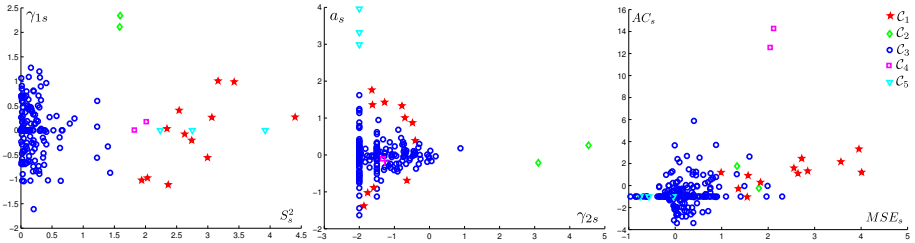

Fig. 3. Clustering space for the six metrics (each point represents a segment)

events are correctly segmented by the algorithm ($\mathcal{C}_1$ and $\mathcal{C}_5$ segments are always close to the DO onset) and that there are not "false positives" labels ($\mathcal{C}_1$ and $\mathcal{C}_5$ segments are not found in a non DO event part of the series). However, five events are not detected: 2, 9, 11, 13 and 16 (some of which have been found in the literature to be caused by random fluctuations of the dynamics of the time series and for which there is no evidence of increase in the selected statistics). The clustering space of this segmentation can be analysed in Fig. 3. This Figure confirms that there are some differences between the two clusters associated to the DO events ($\mathcal{C}_1$ and $\mathcal{C}_5$), mainly from the values of the S_s^2 metric.

4 Conclusions

This work tackles the problem of time series segmentation in the context of paleoclimate time series analysis. We propose a Genetic Algorithm to perform the segmentation and clustering of the time series by using six statistic characteristics that have been found to reveal incoming tipping points (TPs). The results have shown that the method clusters together most of the TPs avoiding "false positives", which is a promising result because it demonstrates that most of the TPs can be segmented with the same data analysis, as opposed to other proposals in the literature, which design TP-specific descriptors. Future work includes

extending the method to find early warning signals and considering other time series datasets, mutation and crossover operators and fitness functions.

References

1. Wassmann, P., Lenton, T.: Arctic tipping points in an earth system perspective. AMBIO 41(1), 1–9 (2012)
2. Allen, M.: Planetary boundaries: Tangible targets are critical. Nature Reports Climate Change, 114–115 (2009)
3. Lenton, T.M.: Early warning of climate tipping points. Nature Climate Change 1(4), 201–209 (2011)
4. Dakos, V., Carpenter, S.R., Brock, W.A., Ellison, A.M., Guttal, V., Ives, A.R., Kefi, S., Livina, V., Seekell, D.A., Van Nes, E.H., et al.: Methods for detecting early warnings of critical transitions in time series illustrated using simulated ecological data. PLoS One 7(7), e41010 (2012)
5. Keogh, E., Chu, S., Hart, D., Pazzani, M.: An online algorithm for segmenting time series. In: Proceedings IEEE International Conference on Data Mining, ICDM 2001, pp. 289–296 (2001)
6. Andersen, K.K., Azuma, N., Barnola, J.M., Bigler, M., Biscaye, P., Caillon, N., Chappellaz, J., Clausen, H.B., Dahl-Jensen, D., Fischer, H., et al.: High-resolution record of northern hemisphere climate extending into the last interglacial period. Nature 431(7005), 147–151 (2004)
7. Tseng, V.S., Chen, C.H., Huang, P.C., Hong, T.P.: Cluster-based genetic segmentation of time series with dwt. Pattern Recognition Letters 30(13), 1190–1197 (2009)
8. Sclove, S.L.: Time-series segmentation: A model and a method. Information Sciences 29(1), 7–25 (1983)
9. Himberg, J., Korpiaho, K., Mannila, H., Tikanmaki, J., Toivonen, H.T.: Time series segmentation for context recognition in mobile devices. In: Proceedings IEEE International Conference on Data Mining, ICDM 2001, pp. 203–210 (2001)
10. Chung, F.L., Fu, T.C., Ng, V., Luk, R.W.: An evolutionary approach to pattern-based time series segmentation. IEEE Transactions on Evolutionary Computation 8(5), 471–489 (2004)
11. Xu, R., Wunsch, D.: Clustering. IEEE Press Series on Computational Intelligence. Wiley (2008)
12. Rand, W.M.: Objective Criteria for the Evaluation of Clustering Methods. Journal of the American Statistical Association 66(336), 846–850 (1971)
13. Hubert, L., Arabie, P.: Comparing partitions. Journal of Classification 2(1), 193–218 (1985)
14. Peterson, L.C., Haug, G.H., Hughen, K.A., Röhl, U.: Rapid changes in the hydrologic cycle of the tropical atlantic during the last glacial. Science 290(5498), 1947–1951 (2000)
15. Svensson, A., Andersen, K.K., Bigler, M., Clausen, H.B., Dahl-Jensen, D., Davies, S.M., Johnsen, S.J., Muscheler, R., Parrenin, F., Rasmussen, S.O., Röthlisberger, R., Seierstad, I., Steffensen, J.P., Vinther, B.M.: A 60 000 year greenland stratigraphic ice core chronology. Climate of the Past 4(1), 47–57 (2008)