Computers & Geosciences 175 (2023) 105342

Contents lists available at ScienceDirect

Computers and Geosciences

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

Terrain methods on spectral analysis for paleoclimate interpretations: A novel visualization technique using python

J. Sánchez-Morales [a,*], E. Pardo-Igúzquiza [b], F.J. Rodríguez-Tovar [a]

[a] Department of Stratigraphy and Paleontology, Campus Fuentenueva, Universidad de Granada, 18071, Granada, Spain
[b] Instituto Geológico y Minero de España (IGME), Ríos Rosas 23, 28003, Madrid, Spain

ARTICLE INFO

Keywords:
Python
Computer imaging
Spectral analysis
Paleoclimate
Visualization

ABSTRACT

Spectral analysis techniques are valuable tools at the disposal of paleoclimate scientists in their research of cyclical phenomena potentially responsible for past climatic fluctuations. Advances in computing and an increased availability of climate time series have helped to consolidate this approach. Yet the visual representation of spectral analysis results has not improved at the same pace. Time-frequency analysis aims to identify periodic signals that vary over time using 2-D color graphs, depicting spectral bands theoretically discernible from the image background. The paleoclimate literature is full of examples such as the continuous wavelet analysis, the evolutionary fast Fourier transform (FFT) spectrogram, or even the more recent synchrosqueezing transform. Our approach is based on the stack of non-evolutive (assuming stationary behaviour) spectral analysis results from thousands of fixed interval time series, previously split from a longer and unevenly spaced (irregular sampling) paleoclimate time series. As illustrative examples, the targeted time series are derived from the LR04 Global Pliocene-Pleistocene Benthic $\delta^{18}O$ stack and from the 65 °N summer insolation for the last 5.3 Myr, by means of the Lomb-Scargle periodogram technique. Enhanced and clearer visualization is achieved through the novel incorporation of the terrain analysis techniques: slope, hillshading and color mapping, and posterior blending of the individual images, using Python code. The result consists of a high resolution graphical output, allowing for better qualitative and quantitative interpretations of the obtained cyclicity, as the code incorporates the import of the achieved confidence levels from the spectral technique and the option to obscure the pixels under a certain threshold value. The application of terrain analysis techniques on visualization of spectral analysis results has the purpose of improving previous graphical representations of paleoclimate time-series, especially those time-frequency aspects of the involved phenomena. New developments of our approach may be applied to time-frequency analysis directly, supporting present and future paleoclimate studies.

1. Introduction

Time series generated from paleoclimate proxy archives (i.e., ice cores, speleothems, etc.) often represent non-stationary physical processes. Therefore, the dominant cyclicities cannot be assumed to be constant neither in time scale nor magnitude. Because they vary over time, their analysis becomes very sensitive to the time interval of choice. Time-frequency analysis can deal with time-varying power spectra (evolutive spectral analysis in Berger et al., 1990), where the one-dimensional time series as input is transformed into a two-dimensional time-frequency representation. Advances in computing have led time-frequency analysis techniques like the wavelet transform (Torrence and Compo, 1998) to often complement traditional techniques —e.g. short-time Fourier transform (STFT) and/or windowed Fourier transform (WFT) (see Welch, 1967) —in studies of the variability of series over time. Wavelet analysis may be applied to diverse fields, such as ecology (Cazelles et al., 2008), hydrology (Maheswaran and Khosa, 2012), paleoecology (Sen et al., 2009), climate variability (Azuara et al., 2020) and astronomical forcing (Nie et al., 2008), among others.

One common characteristic of time series from paleoclimate proxies is uneven sampling, varying from quasi-regular to highly irregular intervals. This circumstance can be overcome by the usage of the Lomb–Scargle periodogram (Lomb, 1976; Scargle, 1982; Press et al., 1992), which can directly use uneven data. Initially, this was a deterrent for using wavelet and/or time-frequency analysis (Scargle, 1997; Sen

* Corresponding author.
E-mail addresses: josesanmor@correo.ugr.es (J. Sánchez-Morales), e.pardo@igme.es (E. Pardo-Igúzquiza), fjrtovar@ugr.es (F.J. Rodríguez-Tovar).

https://doi.org/10.1016/j.cageo.2023.105342
Received 3 April 2022; Received in revised form 23 February 2023; Accepted 30 March 2023
Available online 4 April 2023

et al., 2009). However, the potential of the wavelet transform to deal with unevenly spaced time series has been available (e.g., Foster, 1996), and can be applied in paleoclimate research (i.e., Witt and Schumann, 2005). Another past drawback of the wavelet transform is that it was deemed supportive of results from the qualitative standpoint, but quite often the lack of statistical significance tests limits the quantitative interpretation (Torrence and Compo, 1998), in part because of the complexity of the existing methods (Maraun et al., 2007). Luckily, latest developments such as the WAVEPAL Python package (Lenoir and Crucifix, 2018a, 2018b) have addressed this issue by incorporating significance testing against a large choice of background processes. A further important and challenging aspect of wavelet implementation is the choice of the mother wavelet (Kirby, 2005; Maheswaran and Khosa, 2012).

Amongst the software packages that exist today in the time-frequency domain for estimating and depicting the cyclicity and variability of paleoclimate time series, we cite here three sound examples for reference purposes. Thus, using MATLAB, the so-called Signal Processing Toolbox contains several spectral estimators within the time-frequency analysis section (https://www.mathworks.com/help/signal/time-frequency-analysis.html). One important estimator is the continuous wavelet transform, usually referred as cwt, that has been used for instance at describing the eastern African paleoclimate over the last ~620 kyr (Duesing et al., 2021), or for reconstructing modern high-resolution climate in the South China Sea (Zhao et al., 2021). On the other hand, the WAVELET Python package (https://github.com/guillaumelenoir/WAVEPAL) is able to perform time-frequency analysis of irregularly sampled time series using the smoothed Morlet wavelet scalogram. The package requires Python 2.x, and it has been applied for example to the study of the Mid-to Late-Holocene Mediterranean climate variability (Azuara et al., 2020), and the Indian Summer Monsoon variability (Reddy and Gandhi, 2022). Last but not least, the original wavelet analysis from Torrence and Compo (1998) is available in IDL, Fortran, Python and MATLAB programming languages (http://atoc.colorado.edu/research/wavelets/).

All these time-frequency solutions incorporate the generation of the corresponding 2D plots of the results, but with the only graphical capabilities of using a color ramp together (in some cases) with the generation of contour lines for the significance levels. The present work may challenge the conceptual way by which the results from the above software packages are plotted. The framework where our proof of concept has been implemented is the group of non-evolutive spectral techniques, and one of the advantages of this approach is that the code can be easily tuned to work with spectral results from different non-evolutive spectral software packages, without requiring the learning transition to time-frequency techniques. Future developments may well be applied directly to time-frequency visualizations. This paper describes a novel visualization technique that improves the visual representation of those methods used in the time-frequency analysis so far. The code deals with the visualization challenge by importing hundreds to thousands of previously estimated power spectrum time series, as if they were terrain cross-sections. Each cross-section or transect is representative of a unique moment in time. In geosciences, a cross-section is simply the intersection of a 3D entity with a plane. The goal is to stack the cross-sections from left to right, so that they resemble a two-dimensional terrain elevation dataset from the visual point of view, therefore enhancing the spectral results when plotted against time and frequency. A terrain elevation model, also commonly referred to as a Digital Elevation Model (DEM), is no more than the representation of the topographic surface of the Earth, depicting ridges, valleys, etc.

Each spectral profile becomes this way a line of pixels as it would be contemplated like in a map (from above the ground). The terrain visualization methods that have been integrated in the proposed solution are the techniques known as color-relief, hillshade and slope (i.e., Shapiro et al., 1993). The color-relief typically uses a set of colors to differentiate elevation intervals. The hillshade uses an artificial source of light to project the shadows over the landscape based on the terrain variations. The slope is generated by using the gradient estimates of the terrain dataset to enhance steepness versus flatness. The code allows both a qualitative and quantitative analysis of the spectral results, because the confidence levels generated by the spectral analysis technique of choice, can be imported into the visualization pipeline. Our visualization method complements existing time-frequency spectral methods, such as the continuous wavelet analysis, because it allows the possibility of using traditional estimators of power spectrum, offering a quick and effective re-assessment of spectral results. This output when compared to those of paleoclimate time series produced by time-frequency domain visualizations, offers a higher resolution with greater scientific potential, that ultimately may justify their inclusion as a new visualization feature in time-frequency methods.

2. Methodology

The code aims for the visualization of thousands of contiguous multivariate time series representing power spectral values at different frequencies, which are generated from spectral analysis techniques outside the time-frequency analysis domain (i.e., non-evolutive Lomb-Scargle periodogram), resulting in the type of visualization that one would expect from a time-frequency analysis, capable of analyzing the whole time series directly. There are three considerations of the visualization code to work: (1) the split of the original time series into shorter time series with some (constant) degree of overlapping between consecutive time intervals, (2) the time window used for the split of the time series must be constant, and (3) the output files of the spectral technique of choice need to match the structure of two columns space-delimited, which will be used as the required input files to the visualization code. However, the presented code has the flexibility to be adapted in future releases to meet any other requirements.

We have tested the code by using irregularly spaced time series, common in paleoclimate studies. The interpolation of the data should be avoided as it introduces bias (Schulz and Stattegger, 1997; Rehfeld et al., 2011). Consequently it is recommended to use a spectral method that can deal with the uneven series directly, such as the Lomb-Scargle periodogram. This technique uses an oversampling ratio of frequencies and smoothing, which introduces a loss of degrees of freedom in the estimates because the neighbouring frequencies are highly correlated in the estimated power spectrum (Press et al., 1992; Pardo-Igúzquiza and Rodríguez-Tovar, 2012). The program SLOMBS (Spectral LOMB-Scargle periodogram) addresses this bias by using the permutation test to evaluate confidence levels, meaning that the Lomb-Scargle periodogram is calculated also for a large number (>1000) of random permutations of the time series. Thus, SLOMBS uses white noise as the underlying hypothesis for evaluating the confidence levels. However, the visualization code is independent of the noise model of choice, as it simply displays those power spectra values that are above a minimum achieved confidence level, previously set by the user, as in the case of SLOMBS using white noise as hypothesis.

In other words, the usage of SLOMBS is not a requirement for running the visualization code, and it has been considered as adequate as the because the time series to be analyzed are unevenly sampled. Other alternatives to deal with unevenly spaced time series can potentially be used with our code (i.e. REDFIT in Schulz and Mudelsee (2002)). In its current form, the proposed visualization code becomes not applicable to those time-frequency analysis software packages that do not meet those three pre-requisites. However, there are a number of classical spectral techniques (Blackman-Tukey, Thompson multi-taper, etc.) that can greatly benefit from this new visualization technique.

The method is novel in that it assimilates the three dimensions of time-varying power spectra —time, period (or frequency) and power — in terms of the three dimensions of geographic space, respectively: longitude, latitude and elevation.

The spectral power is treated as the terrain elevation (z axis) or the

third geographic dimension. The association between time and longitude (x-axis), and between period/frequency and latitude (y-axis) is arbitrary, but the key lies in treating the power spectrum as a terrain magnitude. The popular technique of wavelet analysis likewise relies on this plotting configuration: time in the x-axis, period in the y-axis and spectral power in z-scale (as a color pallet). Python is the programming language of choice because of its readability (less verbose than other languages such as Java), and faster implementation (plenty of available material for learning and development resources), but more importantly the existence of the opencv-python (cv2) library allowing the handling of the terrain images using Python. Its growing usage in the geosciences is reflected by the increasing number of papers referring to this programming language in recent years (i.e., Computers & Geosciences: 30 studies during 2016, 28 during 2017, 30 during 2018, 29 during 2019, 46 during 2020, 65 during 2021 and 77 during 2022).

2.1. Terrain analysis

Terrain analysis (Wilson and Gallant, 2000) has evolved rapidly since the advent of software and computers. The widespread development of digital elevation models (DEMs) and Geographic Information Science (Longley et al., 2015) have made the features of terrain analysis accessible to users beyond the domain of geography. The code makes use of the slope, hillshading (or relief shading) and color mapping. These concepts will be introduced next.

Slope estimation can be achieved by calculation algorithms (Jones, 1998) including the so-called average maximum technique (Burrough and McDonell, 1998) used in ArcGIS software. Slope represents the maximum value among the differences between a cell's elevation and its neighbours' elevations; this inclination can be expressed in degrees or percent rise, hence between zero (flat surface) and 90° (vertical surface).

The second feature, hillshading, is a visualization technique commonly applied to DEMs (Horn, 1981) that takes into account the effects upon a terrain of a pre-determined illumination source (e.g., sunlight). The parameters involved in hillshade calculation are the angle of the light source (best practice is to use 315° clockwise from the north) and the light angle above the horizon (between 0 and 90°, but typically 45°). Similarly, slope maps can generally be converted into a shaded slope product by applying a colormap and some vertical exaggeration using GIS software (e.g., Global Mapper). Both the shading of the slope and the hillshading have the primary purpose of reproducing the natural topographic features more realistically.

Color mapping, in turn, is intended to enhance raster data through the application of a set of colors from a pre-determined artificial or natural color ramp (Darbyshire and Jenny, 2017). A further area of research surrounds the blending of different images into one (Kovesi et al., 2014). Thus, blending of the slope, hillshade and colormap can be a most striking way of presenting spectral analysis results.

In generating the slope, hillshade and color mapping for the visualization purposes of this study, we relied on the open-source (MIT license) tools of the GDAL geospatial library (https://gdal.org/). For the specific action of combining the different geospatial layers into the final blend, we used the Python library "opencv-python".

2.2. Spectral analysis

Spectral analysis has a long history in paleoclimate research —particularly for Quaternary times— and an array of techniques and software are available today (Trauth, 2021). It is necessary to remind here that the usage of any spectral technique requires enough understanding of the involved physical phenomena as without a recognizable forcing mechanism, the mere detection of spectral peaks can be potentially derived from simple repetitions or bias introduced by the methodology itself, even when the spectral power at some detected frequency is statistically significant. What we describe here resembles the overlapping segments feature of the time series in the spectrogram method, but our code is not restricted to the Fourier-based methods of the spectrogram method. Moreover, it offers the possibility of displaying the results of any other underlying spectral technique, such as the maximum entropy spectral analysis (MESA) (Burg, 1967; Parzen, 1968) and/or the Lomb-Scargle periodogram (Lomb, 1976; Scargle, 1982) to cite a few.

Although one of the aims of the code is to transform non-evolutive spectral results into a 'wavelet' form style, some thought is required before running it, as the code does produce the visualization, but not the required spectral time series as inputs that have to be generated independently of the code. Particularly important in our approach, it is deciding on the time window for dividing the original time series into fragments or fixed interval time series, because the window size determines the maximum period that can be safely recognized on each transect. This rule of splitting the original time series into subseries to perform sub spectra analysis is known (e.g., Pardo-Igúzquiza and Rodríguez-Tovar, 2006), although it is not necessary in time-frequency methods (i.e., continuous wavelet analysis), to which our visualization code does not apply.

In the geosciences, uneven time series are common (Pardo-Igúzquiza and Rodríguez-Tovar, 2012). In this context, the required step of splitting the original time series (prior to spectral analysis) into fixed intervals will inevitably produce time series having a different number of records per generated series, only in the scenario of using uneven time series as input to the process. Although this is not a limitation for our code, it may stand as an issue for choosing the right spectral method. So, regardless of whether the original time series has regular or irregular sampling, this point must be addressed when deciding which spectral analysis method is best to use. It is mandatory that the number of estimated frequencies on each output file from the spectral method is the same, because this number represents the y-axis dimension of the visualization. Here, the visualization code was tested on multiple regular (containing the same number of frequencies) time series representing power spectra, all obtained from the spectral analysis conducted on multiple uneven time series pre-defined with overlapping fixed intervals. Additionally, the maximum frequency to be evaluated on each transect must be assessed. The so-called "Nyquist frequency" (Shannon, 1948), is defined as twice the sampling frequency for even time series (Papoulis, 1984). For uneven time series other equivalent approximations are available, such as estimating the average sampling or the minimum distance (Pardo-Igúzquiza and Rodríguez-Tovar, 2012). Thus, there is always a dependency between the sampling ratio and the Nyquist limit. From the visualization point of view, this limit or maximum frequency may represent the upper limit of the y-axis.

Another point to be underlined in any spectral analysis interpretation is the separation of the significant cyclicities from the noise introduced by the methodology itself. This can be achieved by running either standard statistical tests or computer-intensive permutation tests (Pardo-Igúzquiza and Rodríguez-Tovar, 2000). Having introduced this factor into the code, we have enabled the usage of a second input file —per time series file— that contains the achieved confidence level (ACL), ranging from 0 to 100 (or from 0.0 to 1.0). Typically, scientific validity is secured when confidence levels exceed the values of 90%, 95%, or 99%. The code adopts this information as the threshold between the displayed power spectrum values and the no-data values.

Even though the code can deal with different non-evolutive spectral techniques, it has been developed for operating with the output files produced by the SLOMBS computer software (Pardo-Igúzquiza and Rodríguez-Tovar, 2012), which uses the Lomb-Scargle periodogram (Pardo-Igúzquiza and Rodríguez-Tovar, 2012, 2013, 2015; Pardo-Igúzquiza and Dowd, 2020). SLOMBS was integrated by means of the Monte Carlo permutation test to evaluate the significance of the registered spectral peaks against a white noise null hypothesis, and it adjusts the statistical significance by smoothing the periodogram. This procedure has been applied in cyclostratigraphic analyses of time series from core and outcrop data, as it permits characterization of cyclic variations mainly related to climate changes at different ranges.

Significant peaks in spectral analysis have thereby been recognized, involving cycles having time scales longer than those of the Milankovitch band (Hernández-Molina et al., 2016), or cycles precisely from the Milankovitch frequency band (Durán et al., 2013; Vegas et al., 2013; Cantalejo and Pickering, 2015; Hernández-Molina et al., 2016; Lofi et al., 2016; Rodríguez-Tovar et al., 2016), cycles belonging to the sub-Milankovitch frequency band at millennial-to centennial-scale cyclicities (Vegas et al., 2013; Jiménez-Espejo et al., 2014; Rodrigo-Gámiz et al., 2014, 2018), and even in the range of cycles per year (Durán et al., 2013; Morgan et al., 2017). The assumption of red noise would operate exactly in the same way as the white noise regarding the visualization, because the code import the results already generated by the spectral technique.

3. Python code

The flow of the program is described in Fig. 1. In addition to the main steps (points 1 to 4), some previous actions need to be considered (Fig. 2). Thus, before running the code, which takes less than a minute on a modern computer (using ~2000 input files representing one time series each), the input files must be created. They consist primarily of time series files containing both the input (step 2 in Fig. 2) and the output (step 3 in Fig. 2) of each spectral analysis performed on every transect that was split from the original time series. Additionally, template files are provided with the code (color templates). In our case the software was tested using as input the results of the program SLOMBS (Pardo-Igúzquiza and Rodríguez-Tovar, 2012). It is possible to use other software and/or classical spectral methods, excluding time-frequency methods, as long as the output file extensions and the structure of the output data files match those of SLOMBS (step 4 in Fig. 2): .prn, .LOM and.ACL for the file extensions of the input data, power spectra and achieved confidence levels, respectively, each containing two space-delimited columns. Thus, for every time series or transect three files are needed. One is the time series containing the time and the values of the physical variable to be analyzed in the spectral analysis (.prn); the second file is the output with the Lomb–Scargle spectrum results (.LOM), which contains the frequencies and the Lomb–Scargle periodogram values; and the third file holds the evaluated frequencies and the corresponding achieved confidence level results (.ACL).

As the visualization code recreates a typical time-frequency analysis visualization by using several individual data files that are stacked together, it is necessary to split the original time series into smaller —but constant— intervals prior to the spectral analysis (step 2 in Fig. 2). The spectral analysis is then carried out after the split and before running the visualization code. The decision on how to split the original dataset into sub-intervals may affect the shape of the output raster generated by the code, since the number of pixels in the x-direction of the visualization represents the number of analyzed time intervals or input files, while the number of pixels in the y-direction reflects the number of evaluated frequencies during the spectral analysis, which is assumed to remain constant on every spectral analysis run. These two facets (number of time series and number of analyzed frequencies) must be balanced such that the output raster will have legible dimensions. There is no technical limitation regarding the value of these two dimensions, but from a practical point of view, the best visual results are obtained when the output raster is rectangular (i.e., 2015 time series as width and 500 frequencies as height makes a ratio of ~4 as per in the case study). The number of time series can be larger than the number of analyzed frequencies, or vice versa, but always up to a certain limit for avoiding the generation of very slim rectangular shapes (i.e., ratio >10). This way, the common exercise in paleoclimate research of evaluating a particular frequency of interest across time becomes more effective in the produced visualization.

On the other hand, the percentage of overlapping between consecutive time series positively affects the time resolution of the visualization, given that the program uses the mean of each time series interval, or center point, for placing every transect at the corresponding time value of the x-axis. Therefore, the larger the overlap of time series, the smaller the time differences between consecutive pixels across the x-axis. The sampling of the original dataset can be uneven before the split,

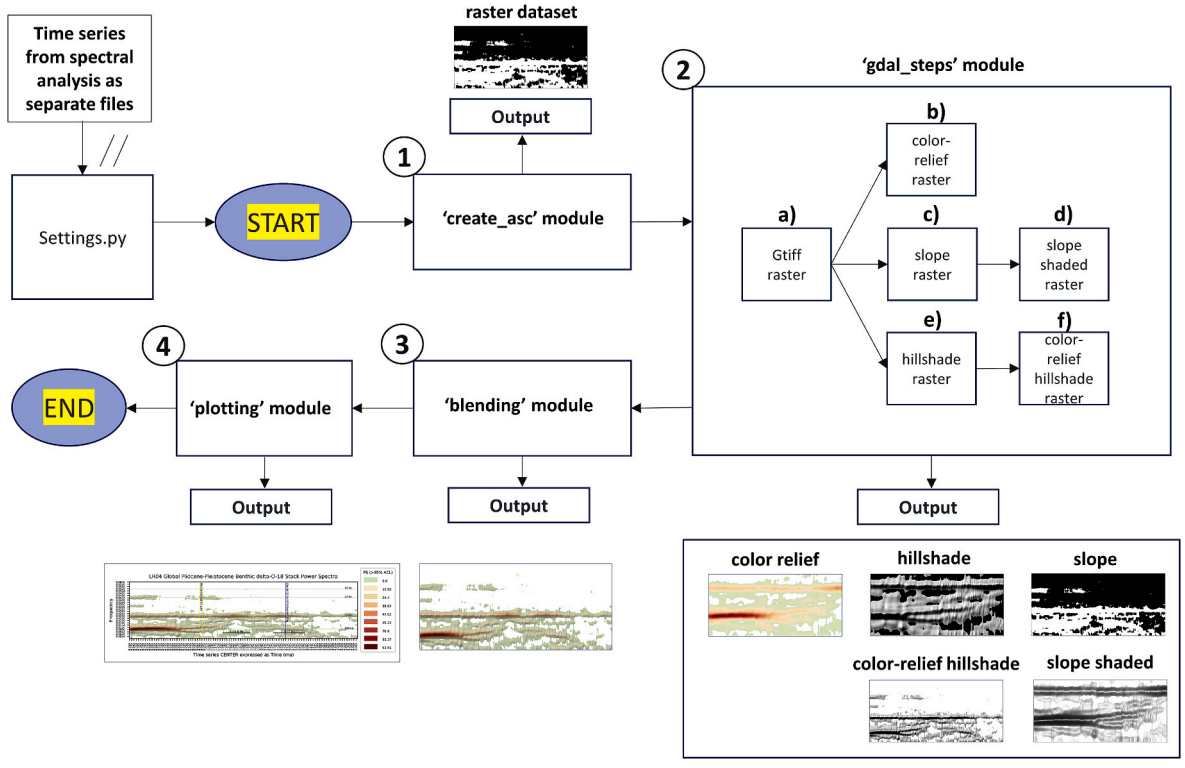

Fig. 1. Code flowchart. Prior to running the code one must have the spectral results ready and the configuration parameters set.

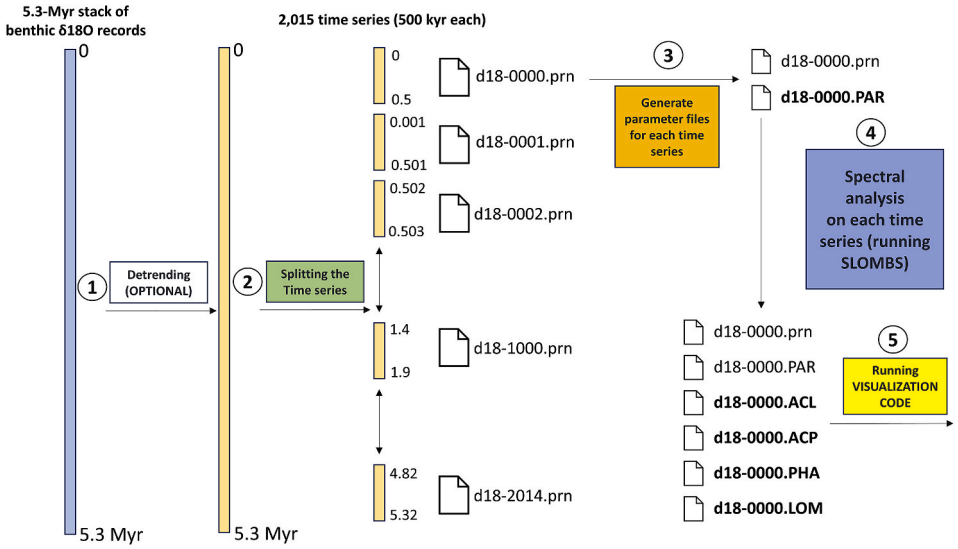

Fig. 2. Sketch of preparatory actions prior to the spectral analysis, describing the split of the original paleoclimate time series and the output files of the spectral analysis, before running the visualization code.

but each new time subseries should be created using the same constant time window. The chosen time window for creating each transect is constant to avoid introducing any temporal heterogeneity along the x-axis, because this way the derived interpretation on the estimated power spectra between consecutive transects is considered to be based on the same time window and the same analyzed frequencies. Overall, we recommend to separate the subseries by small time steps (e.g., one sampling step of the original series) to increase the resolution at detecting time-frequency changes in the visualization. This approach does not require to interpolate any values, but to use the same original time series in the new time subseries. Consequently, in the case of unevenly spaced time series, the output image has irregular time resolution and constant frequency resolution, whereas for evenly spaced time series, both time and frequency resolutions are constant. The handling of the irregular time resolution (for uneven time series) is performed by the visualization code during the plotting phase (step 4 in Fig. 2), so the right values are transferred to the time scale. All the input data files must be placed in the folder specified for this purpose in the settings file.

Once the input data to the code is in place, a few configuration parameters must be set in the settings.py file before running the code (all these parameters can be found in Table A1). When the code is run, the **first module** (step 1 in Fig. 1) 'create_asc' reads all the.prn, .LOM and. ACL input files and generates a 2-D raster file formed by columns (time) and rows (frequencies) in ASCII-format raster (.asc file extension). Just after the.asc file is generated, a projection file.prj is created, where the coordinate system properties are given, so the data becomes geographically enabled and the terrain techniques can be applied. The geographic projection is hard-coded into the projection file, meaning there is no need for the user to deal with it; the British National Grid (BNG) projection (EPSG:27700) is the one we selected to enable the geographic features. The template could be changed to host any other projected coordinate system, but there is no evidence that by doing so the visualization will be improved. Thus, in the BNG every pixel translates into 1 m over the Earth's surface as per the linear unit of this projected coordinate system.

The **second module** (step 2 in Fig. 1) 'gdal_steps', is responsible for generating different raster files to be later blended and used as part of the visualization. The module owes its name to the usage of GDAL (Geospatial Data Abstraction Library) tools, a pre-requisite that needs to be installed in the computer where the code is going to be run (code has been developed and tested using version 3.0.2 of GDAL). An important aspect of the designed pipeline using GDAL is that the size of the raster images (matrices) along the process remains unchangeable. So, for instance, if the pipeline is run using an input matrix formed by 500 pixels (representing frequencies) and 2015 pixels (representing time), all the subsequent color, slope and hillshade images will have the same dimensions. The GDAL tools are called from the Python code, being the tools external to the program and installed in the operating system. This approach is recommended as it is more versatile than installing GDAL as an internal Python library, because in that case the so-called Python bindings may need regular maintenance, or they may stop working when upgrading either Python or the GDAL tools to a newer version.

The first raster file to be created by the module is, in fact, the translation of the.asc file into GeoTIFF format (.tif) (a in step 2 of Fig. 1). The reason behind this transformation is that the spatial properties are preserved within the file itself, which is more efficient, reliable and convenient for the following steps of the pipeline. Once the data is translated into a.tif file, the program will use this file to generate five different output files: the color-relief from the data (b in step 2 of Fig. 1), the slope from the data (c in step 2 of Fig. 1), the color-relief from the slope (d in step 2 of Fig. 1), the hillshade from the data (e in step 2 of Fig. 1), and the color-relief from the hillshade (f in step 2 of Fig. 1). The calculation methods and/or algorithms used in the generation of the color-relief, slope and hillshade images are internal to the GDAL tools. According to the online documentation (https://gdal.org/programs/gdaldem.html), they were derived from previous work by the U.S. ARMY CORPS OF ENGINEERS (Shapiro et al., 1993).

For the first step, the GDAL tool 'gdal_translate' is used (https://gdal.org/programs/gdal_translate.html), whereas the GDAL tool 'gdaldem' is used for the remaining steps, as this sub-program is designed to deal specifically with DEMs. The 'gdal_translate' command-line instruction for translating the.asc file into GeoTIFF format (.tif) is:

(1) "gdal_translate -of GTiff -a_nodata <nodata> -ot Float32 -a_srs EPSG:27700 <input_f> <output_f>"

The <nodata> in (1) represents the value of the NODATA variable from the settings.py file (i.e., −999), <input_f> the.asc filename and <output_f> the.tif filename to be created. Next, the color-relief raster is generated using the GeoTIFF created by 'gdal_translate' as input, together with the provided file as the default color template (.clr), based on a sequential list of nine RGB colors. The color template is configurable, and there is no further requirement, except for the usage of three colors as a minimum: white for the no-data values (i.e., −999), one for

the lowest power spectral value and one for the maximum value. Based on our experience the higher the number of colors provided the clearer the visualization (e.g., nine). The structure of the template consists of the power spectral values, followed by the pixel value (0–255) of the red, green and blue bands respectively. The output file is a 3-band RGB raster file representing the data by means of the color-ramp provided. The creation of the color-relief image is achieved in the program by using the following GDAL command-line instruction which sits inside the 'gdal_steps' module:

(2) "gdaldem color-relief -of GTiff <input_f> <color_path> <output_f>"

The <input_f> parameter in (2) corresponds to the raster GeoTIFF containing the power spectra values in time (x) and frequency (y) form, the <color_path> represents the color template file containing the color-ramp, and the <output_f> indicates the filename of the color-relief GeoTIFF to be generated as output. It follows the slope raster, whose values can vary between 0.0 and 90.0° and are stored as a floating-point

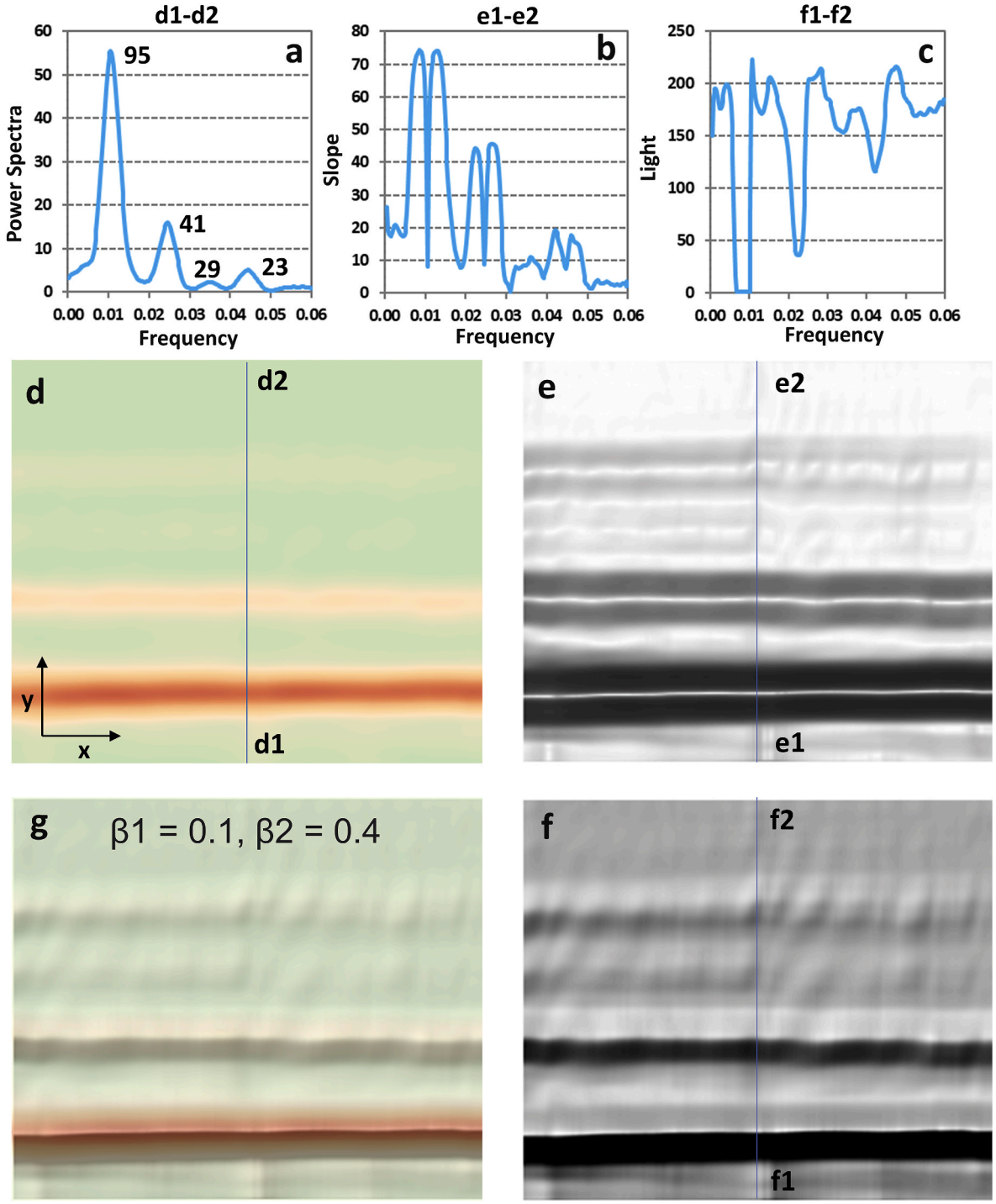

Fig. 3. All GDAL steps contributing to the blending process. Dataset corresponds to the orbitally tuned "LR04" stack time series of benthic $\delta^{18}O$ records (Lisiecki and Raymo, 2005). Transect d1-d2 represents the power spectral values (a), transect e1-e2 represents the slope values and transect f1-f2 represents the hillshade values. Color-relief image (d), slope-shaded image (e), hillshade image (f) and produced final blended image (g) using the default blending parameters of beta1 = 0.1 and beta2 = 0.4. The applied achieved confidence level threshold was set this time to 0 for displaying purposes.

raster file, followed by the shaded slope, which uses a basic color ramp of black (0, 0, 0) for vertical or high slope, and white (255, 255, 255) for flat surfaces. This color ramp is also a file (.txt) available in the 'templates' folder. The generation of the shaded slope is achieved by generating the slope first, followed by applying the color-ramp on it:

(3) "gdaldem slope -of GTiff <input_f> <slope_f>
(4) "gdaldem color-relief -of GTiff <slope_f> <color_path> <slopeshade_f>"

In (3), the <input_f> represents the raster of the power spectra values and the <slope_f> the slope raster filename to be generated, which is used as input in (4). The <color_path> in (4) is the text file containing the color-ramp for the slope and the <slopeshade_f> is the shaded slope filename to be created as output. Next step is the hillshade generation, followed by the shaded hillshade, using an adjusted color ramp which converts the black no-data background of the original hillshade into white. Thus, the shaded hillshade uses the dark color just for the darkest areas that receive few to zero light from the source. These two steps for the hillshade consist of these GDAL instructions:

(5) "gdaldem hillshade -of GTiff <input_f> <hillshade_f>
(6) "gdaldem color-relief -of GTiff <slope_f> <color_path> <shaded_hillhade_f>"

In (5), the <input_f> represents the raster of the power spectra values and the <hillshade_f> the hillshade raster filename to be generated, which is used as input in (6). The <color_path> in (6) is the text file containing the color-ramp for the hillshade and the <shaded_hillhade_f> is the shaded hillshade filename to be created as output.

The **third module** (step 3 in Fig. 1) of the program, the 'blending' module, uses the opencv-python library to merge the color, the slope shaded and the hillshade shaded raster files into a single file. To this end, the color image and the slope shaded are blended first, and the resulting image is then blended with the hillshade raster file. By default, the color image has a weight of 1, and the slope shaded image has a weight of 0.1, followed by the hillshade image with a weight of 0.4. In this context the weight coefficient is equivalent to opacity. These coefficients can be configured in the settings file. All the individual raster files generated during the process can be later examined by the user if the <REMOVE_TEMP > variable is set to False in the settings file, which means that the intermediate files are not removed.

Each generated image, or step, contributes in a different manner to the blended image as illustrated in Fig. 3. Focusing on the component parts of the GDAL pipeline, a random transect along the y-axis (constant age) for each terrain technique best showcases the relative contribution of each applied technique. Thus, the color-relief image (Fig. 3d) uses bands of color to effectively enhance differences in the obtained power spectra results (Fig. 3a). The default color-ramp using 9 RGB values grants the differentiation of the extreme values in the dataset. The shaded-slope image (Fig. 3e) is depicted using black tones for high slope values and white tones for low slope values. The cross-section on the slope raster (Fig. 3b) reveals the contribution of the slope component for defining accurately both the spectral peaks or ridges and the valleys, as both features have a markedly low slope value. The hillshade image (Fig. 3f) eases the visual identification of the elevated areas because they are lightened up in their north face, and shadowed in their south face (Fig. 3c). The blended image (Fig. 3g) combines all the previous features into the same image, aiming for defining the steep spectral peaks plus detailing the morphology of all the other spectral results.

The **fourth** and last module of the program (step 4 in Fig. 1) is the 'plotting' module. It adds optional graphical features such as axes, ticks, labels, titles, and vertical/horizontal lines. This module uses the Python library matplotlib; and although it enhances the final plot around the blended raster file, it does not alter or modify the blended raster produced in step 3, which remains as a separate file in the 'output' folder.

4. Results and discussion: case study

4.1. Orbital forcing and pleistocene glaciation

The Pleistocene (0.01–2.58 Ma) comprises most of the Quaternary period's history, and it offers numerous examples as potential case studies to test our code. It is characterized by the occurrence of glacial and interglacial periods provoking both global ice volume increases and decreases, as well as the associated sea level oscillations. The Pleistocene climate is being investigated and discussed widely.

The current paradigm to explain the alternations between cold glacial periods and warmer interglacial periods was first theorized by Milutin Milankovitch (1941) (see also Maslin, 2016, for a review on this subject). He postulated that reductions of the solar irradiation reaching the Earth's surface —hence the term orbital forcing— during summer at the latitude of 65 °N, could potentially lead to an ice age. Today we refer to the Milankovitch cycles in the geological record as the precession, obliquity and eccentricity cycles. Precession is the combined movement of the Earth's axial precession and the Earth's apsidal precession; the first one (clockwise) is the absolute movement of the Earth's rotation axis describing a cone in space (one cycle takes about 26 kyr) and the second one (counterclockwise) is the precession of the Earth's orbit or the precession of the line connecting the apsides of Sun and Earth (one cycle takes about 112 kyr). Obliquity is the variation (tilt) of the Earth's axis of rotation; though its current value is about 23.44°, it can oscillate between roughly 22.1 and 24.5°. Eccentricity is the change in shape of the Earth's orbit from a perfect circle. The length of these astronomical cycles may vary (Rodríguez-Tovar, 2014): between 19 and 24 kyr for precession (extremes at 14 and 28 kyr), around 41 kyr for obliquity (extremes at 28 and 54 kyr) and between 100 and 400 kyr for eccentricity.

A few decades after the initial theory was formulated, researchers were able to test and link the paleoclimate record with the insolation input (Hays et al., 1976; Imbrie et al., 1992), including the exposure of further scientific issues. The visual representation of some of these issues will be the target of our case study. We selected the '100 kyr problem' (Raymo and Nisancioglu, 2003), the 'Milankovitch 41 kyr problem' (Raymo and Nisancioglu, 2003), the 'eccentricity myth' (Maslin and Ridgwell, 2005) and the 'late Pliocene-early Pleistocene 100 kyr problem' (Nie et al., 2008) to show the utility of our code at presenting a clearer picture of the variations in insolation input over time. Although research on the Pleistocene climate has progressed since those pioneer works, the issues remain largely unresolved, as we explain below.

All four of these issues are interrelated. Assuming that the occurrence of glacial-interglacial periods is controlled primarily by changes in the Earth's orbital parameters (Hays et al., 1976) and that both the 23 kyr (precession) and 41 kyr (obliquity) cycles were found to be direct responses (linear) to high-latitude summer insolation forcing (Imbrie et al., 1992), the '100 kyr problem' is the absence of an astronomical (Milankovitch) explanation for the observed 100 kyr cyclicity of the glacial-interglacial cycles after 0.8 Ma (Raymo and Nisancioglu, 2003). It has been suggested that this 100 kyr cyclicity was caused by internal mechanisms operating within the climate system (Imbrie et al., 1992). For example, the timing of the terminations of the 100 kyr glacial periods was successfully modelled by astronomical variations and CO_2 concentrations (Berger et al., 1999). Precisely, the 'eccentricity myth' represents the view that the '100 kyr problem' is in fact an artifact of spectral analysis, and therefore it is unlikely that the astronomical eccentricity cycle is the main driver of the observed 100 kyr glacial-interglacial periods in the Pleistocene. In turn, the 'Milankovitch 41 kyr problem' evokes the fact that the global ice volume between 0.8 and 3 Ma varied predominantly at the 41 kyr obliquity period, without substantial evidence for the 23 kyr precession period. These two

different orbital forcing scenarios —the '100-kyr world' and the '41-kyr world'— are separated by the Mid-Pleistocene transition (MPT), implying a change in the dominant frequency and mean ice volume, but also a higher contrast between warm and cold periods (Maslin and Ridgwell, 2005). Finally, the 'late Pliocene-early Pleistocene 100 kyr problem' (Nie et al., 2008), refers to the amplitude and phasing mismatch between eccentricity and the ice volume, as well as other paleoclimatic proxy records, between 1 and 3 Ma.

Recent research suggests that obliquity still dominates the '100-kyr world', the sequence of interglacials simply being the deglaciation response when a certain summer insolation threshold was exceeded, which did not happen in every obliquity cycle (Tzedakis et al., 2017). The importance of obliquity in the pacing of the Pleistocene glacial terminations during the '100-kyr world' has been corroborated in speleothems (Bajo et al., 2020). Overall, the causes for the MPT, which changed the cyclicity of the glacial cycles from 41 kyr to 100 kyr, are still under scrutiny (Berends et al., 2021). Furthermore, it is now predicted that the influence of Milankovitch cycles on Earth's climate may be delayed (Maslin, 2016) or even suppressed in the future in the context of current climate change (Caccamo and Magazù, 2021).

4.2. Implementation and results

Three different time series were used to illustrate the code capabilities with regard to these paleoclimate issues. One is the 5.3-Myr stack of benthic δ ^{18}O records (Fig. 4a and b), known as the "LR04" stack, built from 57 global sites and orbitally tuned (Lisiecki and Raymo, 2005). Also, a second version of the "LR04" stack but untuned (Lisiecki, 2010) has been used for testing with a different spectral resolution (reducing the number of evaluated frequencies in the y-axis). The third is the

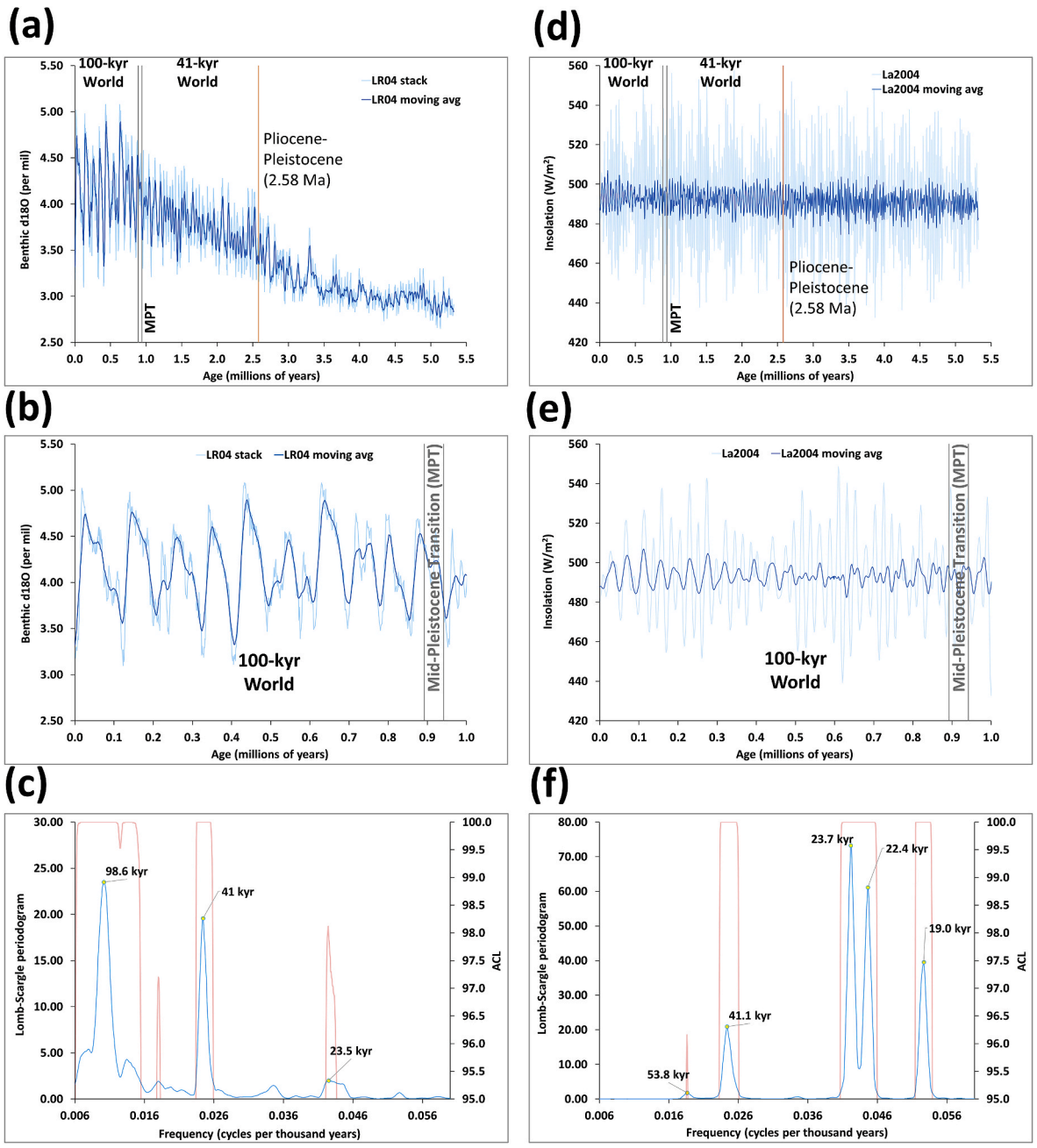

Fig. 4. Time series visualization and Lomb-Scargle spectral results on two datasets: a-c) orbitally tuned "LR04" stack time series of benthic δ ^{18}O records (Lisiecki and Raymo, 2005); d-f) mean daily insolation on June 21 at 65 °N using La2004 orbital solution (Laskar et al., 2004).

computed mean daily insolation on the 21st of June at 65 °N, and for the same period and time intervals of those in the "LR04" stack (Fig. 4d and e), which was obtained using the La2004 orbital solution (Laskar et al., 2004). The insolation dataset was generated to match the irregular time units of the LR04 stack dataset so all the visualizations can be compared along the x-axis (time).

As the primary objective of the case study is to illustrate the evolution of the Milankovitch cycles in the context of the exposed scientific issues, especially across the Pliocene-Pleistocene boundary, the choice of these datasets is well aligned to that goal. Yet, the visualization code relies on the quality of the dataset to be analyzed using spectral analysis.

The first step was to identify the Milankovitch cycles in two of the original time series. The estimation of the Lomb-Scargle periodogram was conducted on both the entire "LR04" stack dataset (Fig. 4c) and the entire daily insolation dataset (Fig. 4f), to test the assumption of stationarity and the detection capabilities of the spectral method over the last 5 Myr. No detrending was applied to the original time series prior to spectral analysis. SLOMBS uses the underlying hypothesis of white noise because of its permutation test as the chosen Monte Carlo evaluation method (Pardo-Igúzquiza and Rodríguez-Tovar, 2012), and this resulted in positive detection of the obliquity and precession Milankovitch cycles for the tuned "LR04" stack and the insolation datasets. For the aforementioned 100 kyr cycle during the Pleistocene it was only detected in the "LR04" stack dataset, all above the 95% level of achieved confidence. The 400 kyr cycle has not been detected in the analyzed Plio-Pleistocene interval, likely caused by the suppression of the climatic response to the long eccentricity cycle by the 100 kyr ice-sheet dynamics (De Boer et al., 2014).

The second step was the split of the time series into smaller time series for all three datasets. It was decided to use a moving window of 500 kyr —with one sampling record or step of separation (varying in the original series between 1 and 5 kyr) between consecutive time series— to generate 2015 fixed interval time series from the original datasets. This choice was motivated by a conservative approach regarding the cycle identification: one-fifth of the length of the series. This interval allows the potential detection of the 100 kyr cyclicity because its length can be contained within the chosen window up to 5 times, where the stationarity assumption was made at orbital time scales.

The choice of input parameters for SLOMBS is based on published recommendations (Pardo-Igúzquiza and Rodríguez-Tovar, 2012). The highest frequency to evaluate should not be higher than the "Nyquist frequency" or its equivalent for unevenly spaced time series. The number of frequencies in the interval represents the pursued level of detail for the obtained power spectra. The number of permutations should be at least ten times the number of evaluated frequencies. But more importantly, a more consistent estimator is obtained by smoothing the periodogram, and the number of smoothing terms can be increased to ensure a neater detection of the main signals, while limiting the appearance of less prominent spectral peaks.

The input parameters chosen for every run of SLOMBS were the same for the tuned "LR04" and insolation datasets: 0.06 as the highest frequency to evaluate (~16.6 kyr), 500 as the number of frequencies in the interval, 5000 as the number of permutations, and 30 as the number of smoothing terms (selecting linear smoothing as option). In regard to the maximum frequency to be analyzed, it is in fact larger than the average sampling of ~2.5 kyr for the "LR04" stack, or equivalent "Nyquist frequency", granting the detection of cycles larger than 16.66 kyr. The 400 kyr or long-eccentricity cycle ($f = 0.0025$) was theoretically in the detection range of the conducted spectral analysis (500 frequencies between 0.0 and 0.06 per transect) but outside the recommended safeguard of one fifth of the length of the series (400 kyr cycle can be potentially contained as many as 1.25 times in the 500 kyr window). In any case, the identification of the 400 kyr cycle has not been obtained (Fig. 5).

These shorter time series contain the original values; no detrending or transformation was performed based on our previous results on the entire series. The achieved detection of the Milankovitch cycles (Figs. 4 and 5) granted the aimed visualization capabilities of our code. However, one alternative is applying some type of detrending on the original time series and perform the spectral analysis on the resulting time series. For example, applying a moving average technique for smoothing the time series, may enhance the frequencies in the lowest part of the spectrum. If the user is interested in the highest frequencies of the spectrum, a possibility is subtracting the smoothed time series to the original time series, and perform the spectral analysis on the obtained residuals.

For visualization purposes, the center or middle point of each time series segment was considered the value to be represented on the x-axis (time axis) as per our methodology. Given that all datasets have time units expressed in kyr and all have uneven sampling (varying between 1 kyr and 5 kyr), the visualization will inevitably have more resolution towards the left of the image, and less to the right, because the time resolution of the data set improved towards the present. As the visualization does not perform any interpolation between transects along the x-axis, the time ticks on the x-axis unequivocally reflect this heterogeneity, and the labels have been adjusted accordingly. The total number of time series analyzed was identical for all three datasets, being 2015 each (6045 total).

For the untuned version of the "LR04" dataset, we changed the setting to a value of 250 for the number of frequencies to evaluate and to a value of 2500 for the number of permutations. The aim was to assess the visualization output by reducing the resolution along the y-axis, but also to test the detection of Milankovitch cycles on the untuned version of the "LR04" stack.

The spectral analysis on every time series segment generated a .LOM extension file (Lomb-Scargle spectrum) and an .ACL extension file (Achieved confidence level) that are inputs to the visualization code. Based on these parameters, the output images had a width of 2015 pixels and a height of 500 pixels for the tuned "LR04" stack and insolation datasets, and a width of 2015 pixels and a height of 250 pixels for the untuned "LR04" stack version. After spectral analysis on the 6045 (3×2015) time series was completed, the visualization code was run four times to produce the four visualizations: "Summer insolation at 65 °N on June 21 Power Spectra" without restrictions on the minimum ACL parameter (Fig. 5a); the same visualization but imposing a minimum value of ACL (above 95%) for displaying the power spectrum (Fig. 5b); the orbitally tuned "LR04 Global Pliocene-Pleistocene Benthic $\delta^{18}O$ Stack Power Spectra" imposing a minimum value of ACL (above 95%) for displaying the power spectrum (Fig. 5c); and the untuned "LR04 Global Pliocene-Pleistocene Benthic $\delta^{18}O$ Stack Power Spectra" imposing a minimum value of ACL (above 95%) for displaying the power spectrum (Fig. 5d).

The generation of the image with no minimum ACL requirement (Fig. 5a) was to prove that the code can effectively suppress the display of any undesired value for the achieved confidence levels and/or to deal with multiple scenarios on this regard (from 0 to 100%). Any power spectral value below the set threshold is represented using the white color (Fig. 5b, c and 5d).

Focusing on the visualization results with the minimum requirement of 95% of ACL, the insolation dataset plot (Fig. 5b) contains a very strong signal in the precession band (19–23 kyr), followed by obliquity (41 kyr). There is no record of the short-eccentricity signal (about 100 kyr, with major components around 97 and 127 kyr) in the power spectrum for the estimated Earth's received insolation (using the Laskar et al., 2004 solution), which was known to be weak (i.e., Imbrie et al., 1992), although it could have been targeted by the usage of a specific band-pass filter (Nie et al., 2008) to enhance its detection.

The results for the orbitally tuned "LR04" stack power spectra (Fig. 5c) clearly delineate a strong 100 kyr cyclicity for most of the last 1 Myr, and a strong obliquity signal at 41 kyr for the last 5 Myr. The results show no detection of the 400 kyr cycle (long-eccentricity). The precession signal is very weak in the tuned "LR04" time series. In the same

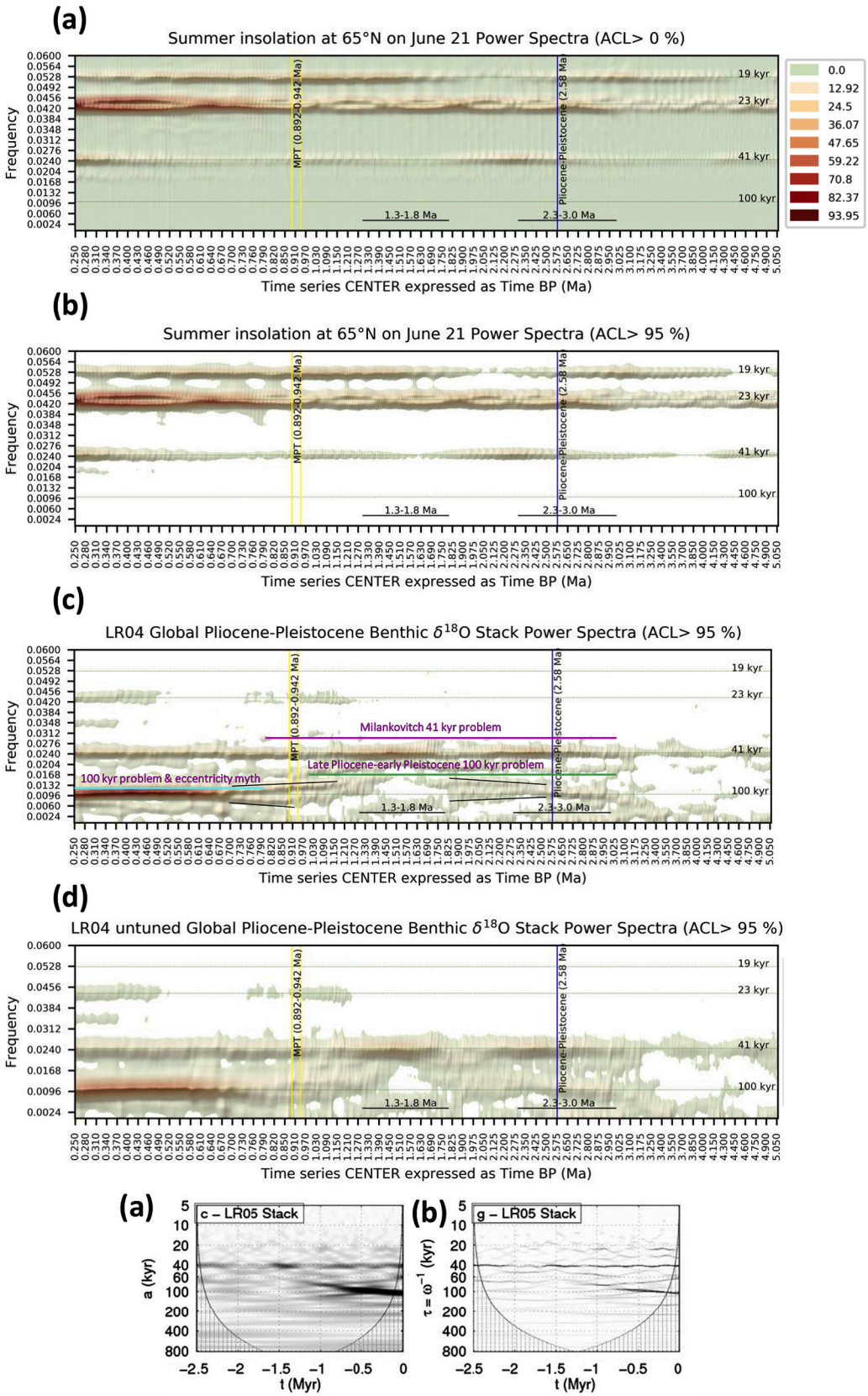

Fig. 5. Visualization outputs from the Python code. a) Summer insolation at 65 °N on June 21 Power Spectra without restrictions on the minimum ACL parameter; b) Summer insolation at 65 °N on June 21 Power Spectra imposing a minimum value of ACL above 95%; c) Orbitally tuned LR04 Global Pliocene-Pleistocene Benthic δ^{18}O Stack Power Spectra imposing a minimum value of ACL above 95%; d) Untuned LR04 Global Pliocene-Pleistocene Benthic δ^{18}O Stack Power Spectra imposing a minimum value of ACL above 95%; e) The LR04 stack using continuous wavelet transform (LR05 in Thakur et al., 2013); f) The LR04 stack using synchrosqueezing time-periodicity decomposition (LR05 in Thakur et al., 2013).

benthic stack, two intervals are well identified in the visualization in the 100 kyr band: one with a strong response to forcing (at 2.3–3.0 Ma) and another one having a weak response (at 1.3–1.8 Ma). These intervals were previously identified (Nie et al., 2008). The paleoclimate frameworks referred to as the '100 kyr problem', the 'Milankovitch 41 kyr problem', the 'eccentricity myth' and the 'late Pliocene-early Pleistocene 100 kyr problem' have been included for visual reference in Fig. 5c.

The plot of the untuned "LR04" stack (Fig. 5d) has on purpose less resolution on the y-axis (250 pixels). This translates into the display of more blurred features becoming generally less sharp than for the tuned dataset version (500 pixels). In any case, the two datasets display almost identical spectral bands around the 100 and 41 kyr cyclicities.

The dominance of the 100 kyr band is clearly visible after approximately 0.69 Ma in both tuned and untuned "LR04" versions. Furthermore, the high resolution of the tuned "LR04" stack allows two new graphical features to be introduced in the paleoclimate discussion, which are less evident in the lower resolution dataset. One feature around the 100 kyr band consists of a broad bimodal characteristic between roughly 0.69 and 1.13 Ma, crossing the MPT boundary (Fig. 5c). Less evident is a second similar feature in the Pliocene, although this time the Y-SHAPED characteristic looks inverted. The second feature is located roughly between 1.82 and 2.58 Ma (both features have been drawn as black lines in Fig. 5c). To put these two visual findings by our code into context, we have brought together two published plots that use the same stack and for the last 2.5 Ma (referred as "LR05" in Thakur et al. (2013)); one using the continuous wavelet transform (Fig. 5e) and the other one using the synchrosqueezing time-periodicity decomposition (Fig. 5f). The synchrosqueezing method has given good results in the time-frequency domain using seismic data (i.e., Wang et al., 2014). In Fig. 5e and f, the first Y-SHAPED feature of the 100 kyr band (~0.69–1.13 Ma) can be identified in a similar position as in our visualization. However, it is the continuous wavelet transform method where this is more evident (Fig. 5e), as the synchrosqueezing method displays many other similar ridges. The second feature (~1.82–2.58 Ma) is not evident in these two plots. The granularity of the referred two methods does not allow a better visual analysis, because of the chosen grayscale color to display the results and poor resolution of the graphical output.

Research on Plio-Pleistocene climate evolution using a flexible change point detection algorithm with sinusoidal models (Ruggieri et al., 2009) postulated that important regime changes can be obtained at 0.78 Ma and 2.7 Ma in the "LR04" dataset, being the other relevant change points at 1.03, 1.2 and 2.45 Ma. All these ages are close to, or within, the depicted Y-SHAPED features in our visualization. On this regard, if considering that the suppression of the 100-kyr glacial cycles during the '41-kyr world' is caused by strong precession forcing disrupting the internal carbon-cycle responses and/or climate feedbacks (Lisiecki, 2010), the existence of the Y-SHAPED features can potentially expose the timings where these regime changes gradually shift from one state to another. In another similar context, looking at a detailed work on insolation forcing leading to interglacials (Tzedakis et al., 2017), the timings of the Y-SHAPED features at 0.69, 1.13, 1.82 and 2.58 Ma closely match the age values of insolation maxima described at 0.694, 1.115, 1.811 and 2.59 Ma, that have been categorised in the indicated study as reaching interglacial thresholds. There are, however, some other relatively close peaks in caloric summer insolation in the study, that are associated with continued interglacial or interstadial status. A close view of the spectral time-frequency evolution using the "LR04" dataset for 0.25–1 Ma (Fig. 6) exposes well the building-up and/or the initiation of the climate events associated with the important 100 kyr cycle during the Pleistocene.

These findings show the potential benefits of applying the exposed visualization method on top of spectral analysis using paleoclimate time series. Yet, the paleoclimate origin and/or forcing mechanism of these new features are beyond the scope of this paper. To further test the usefulness of the proposed visualization on this and/or other aspects of the Quaternary climate research, we recommend expanding its usage using different spectral methods and other paleoclimate time series.

5. Conclusions

The code developed and described here was successfully applied on a major paleoclimate issue: the Pleistocene glaciations. The obtained high-resolution visualization of the Milankovitch cycles across the last 5 Myr validates our novel approach. The potential of the code ultimately relies on the resolution of the time series used, as well as on the capabilities of the chosen spectral analysis technique, but the code alone can improve and refine previous visualizations and their associated interpretations. We also consider this code useful for validating the results derived from time-frequency techniques, such as the wavelet, helping to refine and/or calibrate them, thus providing more accurate results and derived interpretations.

The Python code, hosted in the GitHub platform, guarantees easy access and future updates, while enabling research collaboration.

Authorship contribution statement

Author 1: Conceptualization, main code development, main text writing.

Author 2: Spectral analysis guidance.

Author 3: Case study; Milankovitch cycles and Pleistocene background guidance, text writing (review and editing).

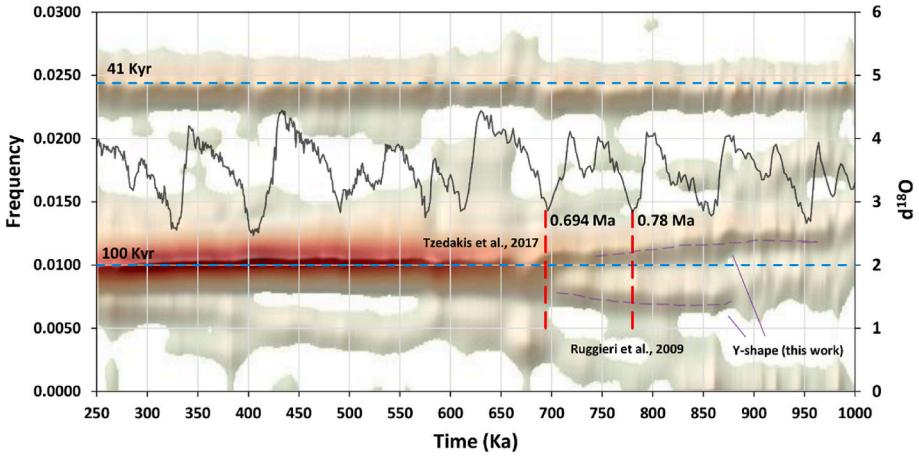

Fig. 6. Visualization of the spectral results from the tuned LR04 stack dataset between 0.25 and 1 Ma, using the same parameters as in Fig. 5c but upgrading the ACL from 95% to 99%. The figure includes both the main 100 and 41 kyr cyclicities (frequencies on the left axis) and the stadial-interstadial variations from the LR04 stack time-series ($\delta^{18}O$ values on the right axis). The tentative start of the dominance of the 100 kyr cyclicity is shown by the location of the Y-SHAPED feature revealed in this present work.

Code availability section

Name of the code: spectral-analysis-visualization-method (MIT license).

Developer: José Sanchez-Morales (josesanmor@correo.ugr.es; jose.chez.morales@gmail.com).

Hardware requirements: Any platform where GDAL utilities are installed (https://gdal.org/).

Program language: tested with Python 3.7.0 on Windows 10.

Software required:

- Python 3 with NumPy (v1.21.2), matplotlib (v3.1.2) and opencv-python (v4.6.0)
- GDAL library tools (tested with v.3.0.2) installed in the computer machine (no need as Python library)

Program size: ~400 KB (without data samples).

The source code is available for downloading at the link: https://github.com/jose-sanchez-morales/spectral-analysis-visualization-method.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

All data samples used in this research have been acknowledged and referenced. Data is openly available at https://github.com/jose-sanchez-morales/spectral-analysis-visualization-method

Acknowledgments

This paper benefited from comments and suggestions by three anonymous reviewers and by Dr. Donner (C&G editor). This work has been supported by Grants PID2019-104625RB-100 funded by MCIN/AEI/10.13039/501100011033 and TED2021-131697B-C21, by FEDER/Junta de Andalucía-Consejería de Economía y Conocimiento/Projects P18-RT-4074, projects B-RNM-072-UGR18 and A-RNM-368-UGR20 (FEDER Andalucía), research groups Research Group RNM-178 (Junta de Andalucía), and Unidad de Excelencia Científica UCE-2016-05 (Universidad de Granada).

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.cageo.2023.105342.

References

Azuara, J., Sabatier, P., Lebreton, V., Jalali, B., Sicre, M.A., Dezileau, L., Bassetti, M.A., Frigola, J., Combourieu Nebout, N., 2020. Mid-to Late-Holocene Mediterranean climate variability: contribution of multi-proxy and multi-sequence comparison using wavelet spectral analysis in the northwestern Mediterranean basin. Earth Sci. Rev. 208, 103232 https://doi.org/10.1016/j.earscirev.2020.103232.

Bajo, P., Drysdale, R.N., Woodhead, J.D., Hellstrom, J.C., Hodell, D., Ferretti, P., Voelker, A.H., Zanchetta, G., Rodrigues, T., Wolff, E., Tyler, J., 2020. Persistent influence of obliquity on ice age terminations since the Middle Pleistocene transition. Science 367 (6483), 1235–1239. https://doi.org/10.1126/science.aaw1114.

Berends, C.J., Köhler, P., Lourens, L.J., van de Wal, R.S.W., 2021. On the cause of the mid-Pleistocene transition. Rev. Geophys. 59, e2020RG000727 https://doi.org/10.1029/2020RG000727.

Berger, A., Melice, J.L., Van Der Mersch, I., 1990. Evolutive spectral analysis of sunspot data over the past 300 years. Phil. Trans. Roy. Soc. Lond. Math. Phys. Sci. 330 (1615), 529–541. https://doi.org/10.1098/rsta.1990.0034.

Berger, A., Li, X.S., Loutre, M.F., 1999. Modelling northern hemisphere ice volume over the last 3 Ma. Quat. Sci. Rev. 18 (1), 1–11. https://doi.org/10.1016/S0277-3791(98)00033-X.

Burg, J.P., 1967. Maximum entropy spectral analysis. Paper presented at the. In: 37th Meeting of the Society of Exploration Geophysicists (Oklahoma City, Oklahoma).

Burrough, P.A., McDonell, R.A., 1998. Principles of Geographical Information Systems. Oxford University Press, New York, p. 190.

Caccamo, M.T., Magazù, S., 2021. On the breaking of the Milankovitch cycles triggered by temperature increase: the stochastic resonance response. Climate 9 (4), 67. https://doi.org/10.3390/cli9040067.

Cantalejo, B., Pickering, K.T., 2015. Orbital forcing as principal driver for fine-grained deep-marine siliciclastic sedimentation, Middle-Eocene Ainsa Basin, Spanish Pyrenees. Palaeogeogr. Palaeoclimatol. Palaeoecol. 421, 24–47. https://doi.org/10.1016/j.palaeo.2015.01.008.

Cazelles, B., Chavez, M., Berteaux, D., Ménard, F., Vik, J.O., Jenouvrier, S., Stenseth, N. C., 2008. Wavelet analysis of ecological time series. Oecologia 156 (2), 287–304. https://doi.org/10.1007/s00442-008-0993-2.

Darbyshire, J.E., Jenny, B., 2017. Natural-color maps via coloring of bivariate grid data. Comput. Geosci. 106, 130–138. https://doi.org/10.1016/j.cageo.2017.06.004.

De Boer, B., Lourens, L.J., Van De Wal, R.S., 2014. Persistent 400,000-year variability of Antarctic ice volume and the carbon cycle is revealed throughout the Plio-Pleistocene. Nat. Commun. 5 (1), 2999. https://doi.org/10.1038/ncomms3999.

Duesing, W., Kaboth-Bahr, S., Asrat, A., Cohen, A.S., Foerster, V., Lamb, H.F., Schaebitz, F., Trauth, M.H., Viehberg, F., 2021. Changes in the cyclicity and variability of the eastern African paleoclimate over the last 620 kyrs. Quat. Sci. Rev. 273, 107219 https://doi.org/10.1016/j.quascirev.2021.107219.

Durán, J.J., Pardo-Igúzquiza, E., Robledo, P.A., López-Martínez, J., 2013. Ciclicidad en espeleotemas: ¿qué señales climáticas registran? Bol. Geol. Min. 124 (2), 307–321.

Foster, G., 1996. Wavelets for period analysis of unevenly sampled time series. Astron. J. 112, 1709–1729. https://doi.org/10.1086/118137.

Hays, J.D., Imbrie, J., Shackleton, N.J., 1976. Pacemaker of the ice ages. Science 194 (4270), 1121–1132. https://doi.org/10.1126/science.194.4270.1121.

Hernández-Molina, F.J., Sierro, F.J., Llave, E., Roque, C., Stow, D.A.V., Williams, T., Lofi, J., Van der Schee, M., Arnáiz, A., Ledesma, S., Rosales, C., 2016. Evolution of the Gulf of Cadiz margin and southwest Portugal contourite depositional system: tectonic, sedimentary and paleoceanographic implications from IODP expedition 339. Mar. Geol. 377, 7–39. https://doi.org/10.1016/j.margeo.2015.09.013.

Horn, B.K., 1981. Hill shading and the reflectance map. Proc. IEEE 69 (1), 14–47. https://doi.org/10.1109/PROC.1981.11918.

Imbrie, J., Boyle, E.A., Clemens, S.C., Duffy, A., Howard, W.R., Kukla, G., Kutzbach, J., Martinson, D.G., McIntyre, A., Mix, A.C., Molfino, B., 1992. On the structure and origin of major glaciation cycles 1. Linear responses to Milankovitch forcing. Paleoceanography 7 (6), 701–738. https://doi.org/10.1029/92PA02253.

Jiménez-Espejo, F.J., García-Alix, A., Jiménez-Moreno, G., Rodrigo-Gámiz, M., Anderson, R.S., Rodríguez-Tovar, F.J., Martínez-Ruiz, F., Giralt, S., Huertas, A.D., Pardo-Igúzquiza, E., 2014. Saharan aeolian input and effective humidity variations over western Europe during the Holocene from a high altitude record. Chem. Geol. 374, 1–12. https://doi.org/10.1016/j.chemgeo.2014.03.001.

Jones, K.H., 1998. A comparison of algorithms used to compute hill slope as a property of the DEM. Comput. Geosci. 24 (4), 315–323. https://doi.org/10.1016/S0098-3004(98)00032-6.

Kirby, J.F., 2005. Which wavelet best reproduces the Fourier power spectrum? Comput. Geosci. 31 (7), 846–864. https://doi.org/10.1016/j.cageo.2005.01.014.

Kovesi, P., Holden, E.J., Wong, J., 2014. Interactive multi-image blending for visualization and interpretation. Comput. Geosci. 72, 147–155. https://doi.org/10.1016/j.cageo.2014.07.010.

Laskar, J., Robutel, P., Joutel, F., Gastineau, M., Correia, A.C.M., Levrard, B., 2004. A Long-term numerical solution for the insolation quantities of the Earth. Astron. Astrophys. 428, 261–285. https://doi.org/10.1051/0004-6361:20041335.

Lenoir, G., Crucifix, M., 2018a. A general theory on frequency and time–frequency analysis of irregularly sampled time series based on projection methods – Part 1: frequency analysis. Nonlinear Process Geophys. 25 (1), 145–173. https://doi.org/10.5194/npg-25-145-2018.

Lenoir, G., Crucifix, M., 2018b. A general theory on frequency and time–frequency analysis of irregularly sampled time series based on projection methods – Part 2: extension to time–frequency analysis. Nonlinear Process Geophys. 25 (1), 175–200. https://doi.org/10.5194/npg-25-175-2018.

Lisiecki, L.E., Raymo, M.E., 2005. A Pliocene-Pleistocene stack of 57 globally distributed benthic $\delta^{18}O$ records. Paleoceanography 20 (1), PA1003. https://doi.org/10.1029/2004PA001071.

Lisiecki, L.E., 2010. Links between eccentricity forcing and the 100,000-year glacial cycle. Nat. Geosci. 3 (5), 349–352. https://doi.org/10.1038/ngeo828.

Lofi, J., Voelker, A.H.L., Ducassou, E., Hernández-Molina, F.J., Sierro, F.J., Bahr, A., Galvani, A., Lourens, L.J., Pardo-Igúzquiza, E., Pezard, P., Rodríguez-Tovar, F.J., Williams, T., 2016. Quaternary chronostratigraphic framework and sedimentary processes for the gulf of cadiz and Portuguese contourite depositional systems derived from natural gamma ray records. Mar. Geol. 377, 40–57. https://doi.org/10.1016/j.margeo.2015.12.005.

Lomb, N.R., 1976. Least-squares frequency analysis of unequally spaced data. Astrophys. Space Sci. 39 (2), 447–462. https://doi.org/10.1007/BF00648343.

Longley, P.A., Goodchild, M.F., Maguire, D.J., Rhind, D.W., 2015. Geographic Information Science and Systems, fourth ed. John Wiley & Sons, p. 496.

Maraun, D., Kurths, J., Holschneider, M., 2007. Nonstationary Gaussian processes in wavelet domain: synthesis, estimation, and significance testing. Phys. Rev. 75 (1), 016707 https://doi.org/10.1103/PhysRevE.75.016707.

Maheswaran, R., Khosa, R., 2012. Comparative study of different wavelets for hydrologic forecasting. Comput. Geosci. 46, 284–295. https://doi.org/10.1016/j.cageo.2011.12.015.

Maslin, M.A., Ridgwell, A.J., 2005. Mid-Pleistocene revolution and the 'eccentricity myth. Geological Society, London, Special Publications 247 (1), 19–34. https://doi.org/10.1144/GSL.SP.2005.247.01.02.

Maslin, M., 2016. Forty years of linking orbits to ice ages. Nature 540 (7632), 208–209. https://doi.org/10.1038/540208a.

Milankovitch, M., 1941. Kanon der Erdbestrahlung und seine Anwendung auf das Eiszeiten-problem. In: Acad, Sp, Serbe, Royale (Eds.), Belgrade (English Translation: Canon of Insolation and Ice Age Problem, Israel Program for Scientific Translation. U.S. Department of Commerce and the National Science Foundation, p. 633 published for the.

Morgan, P., Smrekar, S.E., Lorenz, R., Grott, M., Kroemer, O., Müller, N., 2017. Potential effects of surface temperature variations and disturbances and thermal convection on the Mars in Sight HP3 heat-flow determination. Space Sci. Rev. 211 (1), 277–313. https://doi.org/10.1007/s11214-017-0388-y.

Nie, J., King, J., Fang, X., 2008. Late Pliocene-early Pleistocene 100-ka problem. Geophys. Res. Lett. 35 (21), L21606 https://doi.org/10.1029/2008GL035265.

Papoulis, A., 1984. Probability, Random Variables and Stochastic Processes, second ed. McGraw-Hill International, Singapore, p. 576.

Pardo-Igúzquiza, E., Dowd, P.A., 2020. Maximum entropy spectral analysis of uneven time series in the geosciences. Bol. Geol. Min. 131 (2), 325–337. https://doi.org/10.21701/bolgeomin.131.2.007.

Pardo-Igúzquiza, E., Rodríguez-Tovar, F.J., 2000. The permutation test as a non-parametric method for testing the statistical significance of power spectrum estimation in cyclostratigraphic research. Earth Planet Sci. Lett. 181 (1–2), 175–189. https://doi.org/10.1016/S0012-821X(00)00191-6.

Pardo-Igúzquiza, E., Rodríguez-Tovar, F.J., 2006. Maximum entropy spectral analysis of climatic time series revisited: assessing the statistical significance of estimated spectral peaks. J. Geophys. Res. Atmos. 111 (D10) https://doi.org/10.1029/2005JD006293. D10102.

Pardo-Igúzquiza, E., Rodríguez-Tovar, F.J., 2012. Spectral and cross-spectral analysis of uneven time series with the smoothed Lomb–Scargle periodogram and Monte Carlo evaluation of statistical significance. Comput. Geosci. 49, 207–216. https://doi.org/10.1016/j.cageo.2012.06.018.

Pardo-Igúzquiza, E., Rodríguez-Tovar, F.J., 2013. Análisis espectral de series temporales de variables geológicas con muestreo irregular. Bol. Geol. Min. 124 (2), 323–337.

Pardo-Igúzquiza, E., Rodríguez-Tovar, F.J., 2015. Comparison of the performance of two advanced spectral methods for the analysis of times series in paleoceanography. J. Mar. Sci. Eng. 3 (3), 957–967. https://doi.org/10.3390/jmse3030957.

Parzen, E., 1968. Statistical Spectral Analysis (Single Channel Case) in 1968. Stanford Univ. Press, Stanford, Calif.

Press, H.W., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P., 1992. Numerical Recipes in Fortran, second ed. Cambridge University Press, New York, p. 963.

Raymo, M.E., Nisancioglu, K.H., 2003. The 41 kyr world: milankovitch's other unsolved mystery. Paleoceanography 18 (1), 1011. https://doi.org/10.1029/2002PA000791.

Reddy, P., Gandhi, N., 2022. Indian Summer Monsoon variability on different timescales and deciphering its oscillations from irregularly spaced paleoclimate data using different spectral techniques. In: Holocene Climate Change and Environment. Elsevier, pp. 339–368. https://doi.org/10.1016/B978-0-323-90085-0.00025-5.

Rehfeld, K., Marwan, N., Heitzig, J., Kurths, J., 2011. Comparison of correlation analysis techniques for irregularly sampled time series. Nonlinear Process Geophys. 18 (3), 389–404. https://doi.org/10.5194/npg-18-389-2011.

Rodrigo-Gámiz, M., Martínez-Ruiz, F., Rodríguez-Tovar, F.J., Jiménez-Espejo, F.J., Pardo-Igúzquiza, E., 2014. Millennial-to centennial-scale climate periodicities and forcing mechanisms in the westernmost Mediterranean for the past 20,000 yr. Quat. Res. 81 (1), 78–93. https://doi.org/10.1016/j.yqres.2013.10.009.

Rodrigo-Gámiz, M., Martínez-Ruiz, F., Rodríguez-Tovar, F.J., Pardo-Igúzquiza, E., Ortega-Huertas, M., 2018. Appraising timing response of paleoenvironmental proxies to the Bond cycle in the western Mediterranean over the last 20 kyr. Clim. Dynam. 50 (7), 2925–2934. https://doi.org/10.1007/s00382-017-3782-y.

Rodríguez-Tovar, F.J., 2014. Orbital climate cycles in the fossil record: from semidiurnal to million-year biotic responses. Annu. Rev. Earth Planet Sci. 42, 69–102. https://doi.org/10.1146/annurev-earth-120412-145922.

Rodríguez-Tovar, F.J., Pardo-Igúzquiza, E., Reolid, M., Bartolini, A., 2016. Spectral analysis of Toarcian sediments from the Valdorbia Section (Umbria-Marche-Apennines): the astronomical input in the foraminiferal record. Riv. Ital. Paleontol. Stratigr. 122 (2), 45–60.

Ruggieri, E., Herbert, T., Lawrence, K.T., Lawrence, C.E., 2009. Change point method for detecting regime shifts in paleoclimatic time series: application to $\delta^{18}O$ time series of the Plio-Pleistocene. Paleoceanography 24 (1), PA1204. https://doi.org/10.1029/2007PA001568.

Scargle, J.D., 1982. Studies in astronomical time series analysis. II-Statistical aspects of spectral analysis of unevenly spaced data. Astrophys. J. 263, 835–853.

Scargle, J.D., 1997. Wavelet methods in astronomical time series analysis. In: Rao, T.S., Priestley, M.B., Lessi, O. (Eds.), Applications of Time Series Analysis in Astronomy and Meteorology. Chapman & Hall, New York, NY, pp. 226–248.

Schulz, M., Stattegger, K., 1997. SPECTRUM: spectral analysis of unevenly spaced paleoclimatic time series. Comput. Geosci. 23 (9), 929–945. https://doi.org/10.1016/S0098-3004(97)00087-3.

Schulz, M., Mudelsee, M., 2002. REDFIT: estimating red-noise spectra directly from unevenly spaced paleoclimatic time series. Comput. Geosci. 28 (3), 421–426. https://doi.org/10.1016/S0098-3004(01)00044-9.

Sen, A.K., Filippelli, G.M., Flores, J.A., 2009. An application of wavelet analysis to paleoproductivity records from the Southern Ocean. Comput. Geosci. 35 (7), 1445–1450. https://doi.org/10.1016/j.cageo.2008.10.003.

Shannon, C.E., 1948. A mathematical theory of communication. The Bell System Technical Journal 27 (3), 379–423. https://doi.org/10.1002/j.1538-7305.1948.tb01338.x.

Shapiro, M., Waupotitsch, O., Larson, M., Westervelt, J., 1993. GRASS 4.1 Reference Manual. U.S. Army CERL, U.S. Army Corps of Engineers, Construction Engineering Research Laboratories, Champaign, Illinois, p. 425.

Thakur, G., Brevdo, E., Fučkar, N.S., Wu, H.T., 2013. The synchrosqueezing algorithm for time-varying spectral analysis: robustness properties and new paleoclimate applications. Signal Process. 93 (5), 1079–1094. https://doi.org/10.1016/j.sigpro.2012.11.029.

Torrence, C., Compo, G.P., 1998. A practical guide to wavelet analysis. Bull. Am. Meteorol. Soc. 79 (1), 61–78. https://doi.org/10.1175/1520-0477(1998)079<0061:APGTWA>2.0.CO, 2.

Trauth, M.H., 2021. Spectral analysis in Quaternary sciences. Quat. Sci. Rev. 270, 107157 https://doi.org/10.1016/j.quascirev.2021.107157.

Tzedakis, P.C., Crucifix, M., Mitsui, T., Wolff, E.W., 2017. A simple rule to determine which insolation cycles lead to interglacials. Nature 542 (7642), 427–432. https://doi.org/10.1038/nature21364.

Vegas, J., Pardo-Igúzquiza, E., Galán, L., García-Cortés, A., 2013. Ciclicidad en el registro sedimentario de la laguna del maar de Fuentillejo: historia climática de los últimos 47000 años cal BP. Bol. Geol. Min. 124 (2), 217–234.

Wang, P., Gao, J., Wang, Z., 2014. Time-frequency analysis of seismic data using synchrosqueezing transform. Geosci. Rem. Sens. Lett. IEEE 11 (12), 2042–2044. https://doi.org/10.1109/LGRS.2014.2317578.

Welch, P., 1967. The use of fast Fourier transform for the estimation of power spectra: a method based on time averaging over short, modified periodograms. IEEE Trans. Audio Electroacoust. 15 (2), 70–73. https://doi.org/10.1109/TAU.1967.1161901.

Wilson, J.P., Gallant, J.C. (Eds.), 2000. Terrain Analysis: Principles and Applications. John Wiley & Sons, p. 512.

Witt, A., Schumann, A.Y., 2005. Holocene climate variability on millennial scales recorded in Greenland ice cores. Nonlinear Process Geophys. 12 (3), 345–352. https://doi.org/10.5194/npg-12-345-2005.

Zhao, N., Yan, H., Yang, Y., Liu, C., Ma, X., Wang, G., Zhou, P., Wen, H., Qu, X., Dodson, J., 2021. A 23.7-year long daily growth rate record of a modern giant clam shell from South China Sea and its potential in high-resolution paleoclimate reconstruction. Palaeogeogr. Palaeoclimatol. Palaeoecol. 583, 110682 https://doi.org/10.1016/j.palaeo.2021.110682.