

MyDTW – Dynamic Time Warping program for stratigraphical time series (v. 1.0).

Sergey Kotov, MARUM, 2016

1. Introduction

One of the general tasks in many geological disciplines is matching of one time or space signal to another. It can be classical correlation between two cores or cross-sections in sedimentology or marine geology. For example, tuning a paleoclimatic signal to a target curve, driven by variations in the astronomical parameters, is a powerful technique to construct accurate time scales. However, these methods can be rather time-consuming and can take ours of routine work even with the help of special semi-automatic software.

Therefore, different approaches to automate the processes have been developed during last decades. Some of them are based on classical statistical cross-correlations such as the ‘Correlator’ after Olea [1]. Another ones use modern ideas of dynamic programming. A good example is as an algorithm developed by Lisiecki and Lisiecki [2] or dynamic time warping based algorithm after Pälke [3].

We introduce here an algorithm and computer program, which are also stemmed from the Dynamic Time Warping algorithm class. Unlike the algorithm of Lisiecki and Lisiecki, MyDTW does not lean on a set of penalties to follow geological logics, but on a special internal structure and specific constrains. It differs also from [3] in basic ideas of implementation and constrains design.

2. Dynamic Time Warping: general principles.

The main task of the algorithm is to warp a data sequence X in such a way that the distance between warped and target sequences Y would be minimal, Fig. 1. The algorithm consists of three main steps: 1. The so-called cost matrix calculation, which is simply a matrix of distances between all possible pairs of points in two sequences; 2. Accumulate cost matrix calculation based on dynamic programming according to formulated rules; 3. Optimal warping path finding, Fig. 2.

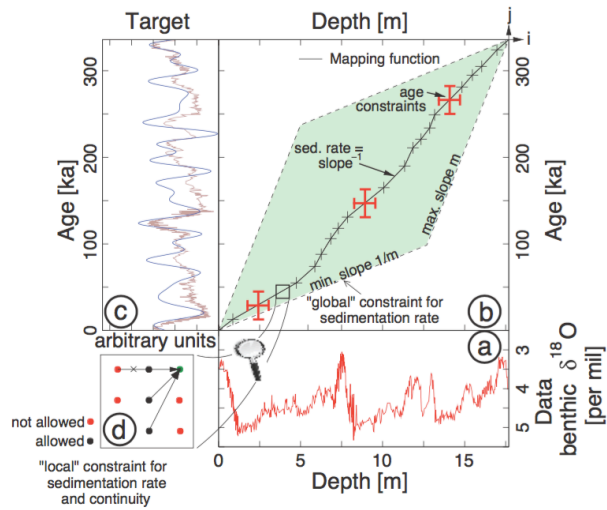


Fig. 1. Illustration on how dynamic time warping generates a mapping function from depth to time, comparing a time series of oxygen isotopes with a calculated target curve of Earth's insolation, subject to sed. Rate and biostratigraphic constraints (from [3]).

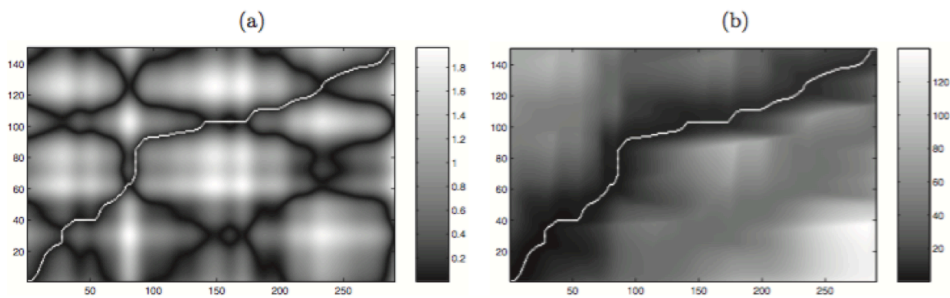


Fig. 2. Cost matrix (a) and accumulate cost matrix (b) with optimal warping path (from [4]).

Rules for the accumulate cost matrix are introduced by local and global constraints which can be specific depending on the task. Details of the algorithm are beyond the scope of this paper.

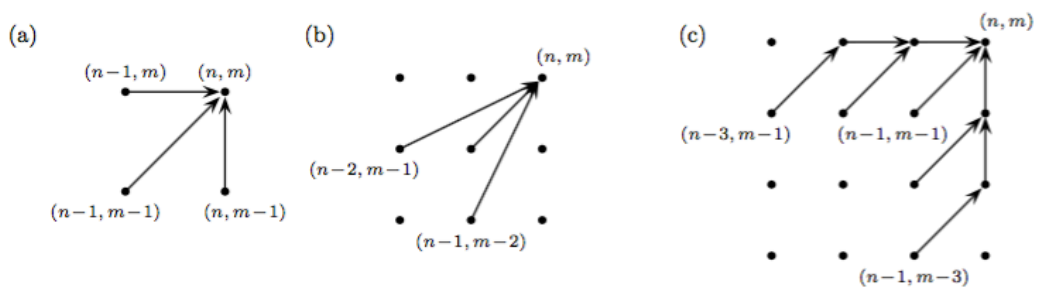


Fig. 3. Examples of local constraints (from [4]).

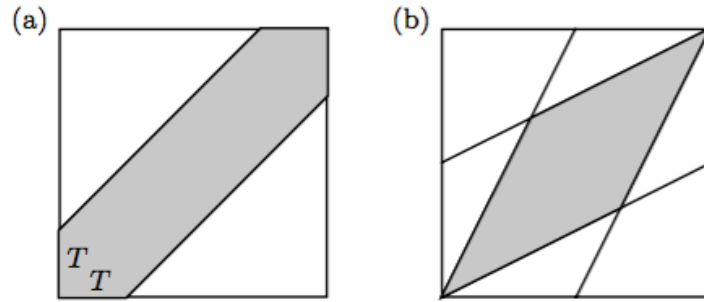


Fig. 4. Examples of global constraints: Sakoe-Chiba band of width T , (b) Itakura parallelogram (from [4]).

Unfortunately, these algorithms cannot be applied to geological data in its traditional form without degradation of signals. Thus, a single element of one sequence may be assigned to several consecutive elements of the other one, leading in geological terms to zero and infinite sedimentation rates what is absurd from the geological point of view, Fig. 5.

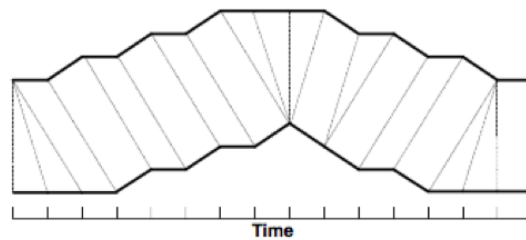


Fig. 5. 'Degradation' of signals with classical DTW technique.

We develop an algorithm avoiding such drawbacks and following special requirements.

3. MyDTW: description of the algorithm.

Requirements to the algorithm:

- No degradation in matched sequences:
 - ✓ Every sample in data must have one and only one link (age) to target
 - ✓ No any of two data samples have the same link (age) to target
- Tuning is controlled by native geological parameters (e.g. Min-Max sedimentation rates)

To satisfy the requirements, the algorithm has two features: 1. Target sequence must be oversampled according to the maximal sedimentation rate; 2. Specific ‘global’ and ‘local’ constrains dynamically driven by reliable sedimentation rates must be applied.

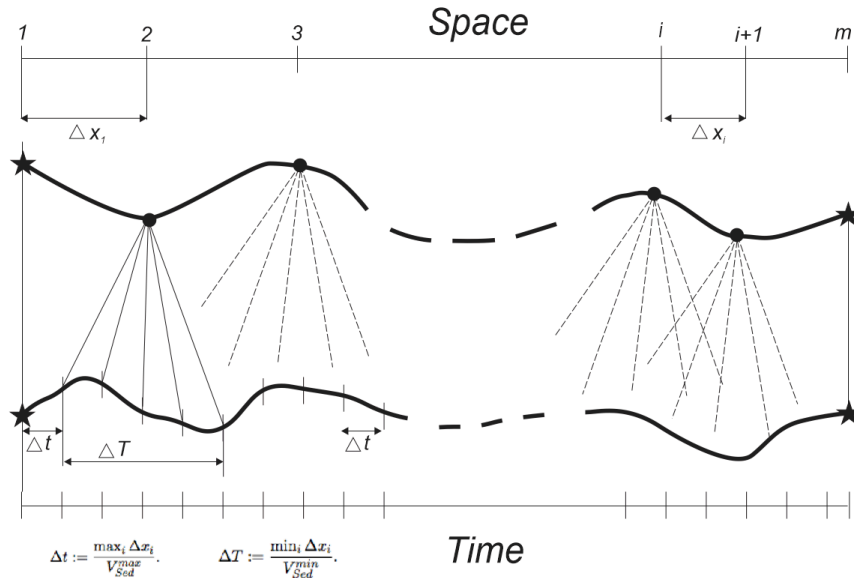


Fig. 6. Target resampling schema for MyDTW.

For the clarity’s sake, let’s consider a physical analogy - a set of ‘bobs’ (‘beans’) connected consequently in a line by springs in a garland (chaplet). According to physical features, any spring can be squeezed - stretched only in a finite extent. In our case, these features are defined by sedimentation rates. So, the task is two redistribute ‘bobs’ from space to time without damage of the spring line. Two limit cases of distribution of ‘bobs’ on the line are shown on the bottom of the figure 7. All allowed paths lie between these constrains. Local constrains are also shown in the middle part of the figure. Global and local constrains change dynamically according to selected sedimentation rates.

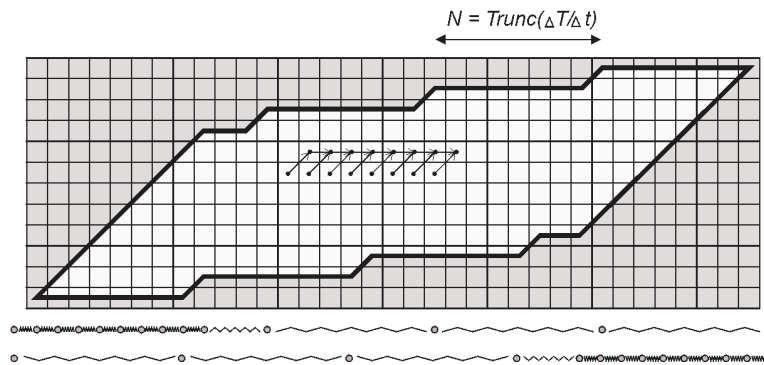
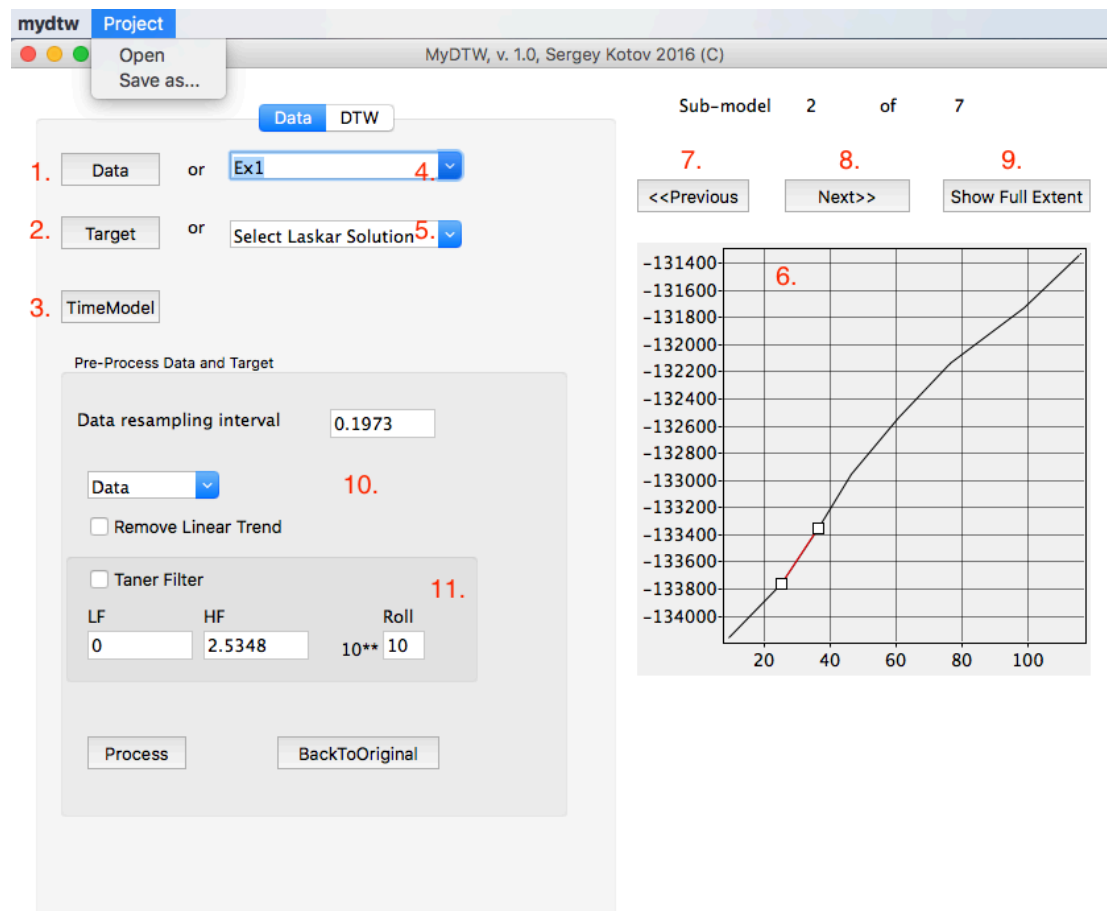


Fig. 7. Global and local constrains of MyDTW.

4. MyDTW: Graphical User Interface (GUI)

The algorithm is implemented as a computer program with a graphical user interface using Free Pascal and Lazarus IDE and available for Windows, Mac OS and Linux (many thanks to the Free Pascal and Lazarus community, <http://www.lazarus-ide.org>). Data and target can be de-trended, pre-filtered with Taner's filter, data can be also resampled with desired resolution prior the tuning. User controls the results with graphical plots: rough time scale with possible multiple tie points, data, target (in black) and warped data (in red), sedimentation rate, cost and accumulate cost matrixes, and optimal warping path in white. Main driving parameters are: data resampling interval, Min and Max sedimentation rates. Additionally, the cost matrix can be calculated not for single measurements but also between windows in data and target in multi-dimensional space. The correlation instead of Euclidian distance is offered for noised signals. There are extra options introduced to restrict the elasticity of the warping path or the derivative of the sedimentation rate, see Fig. 8 and 9. The elasticity parameter mimics the restrictions on the first derivative of the sedimentation rate, i. e. the maximal ratios of two consecutive steps in the best warping path. 'Allow Hiatuses' removes restrictions on minimal sedimentation rate, the warping path is searched based only on the accumulate cost matrix.



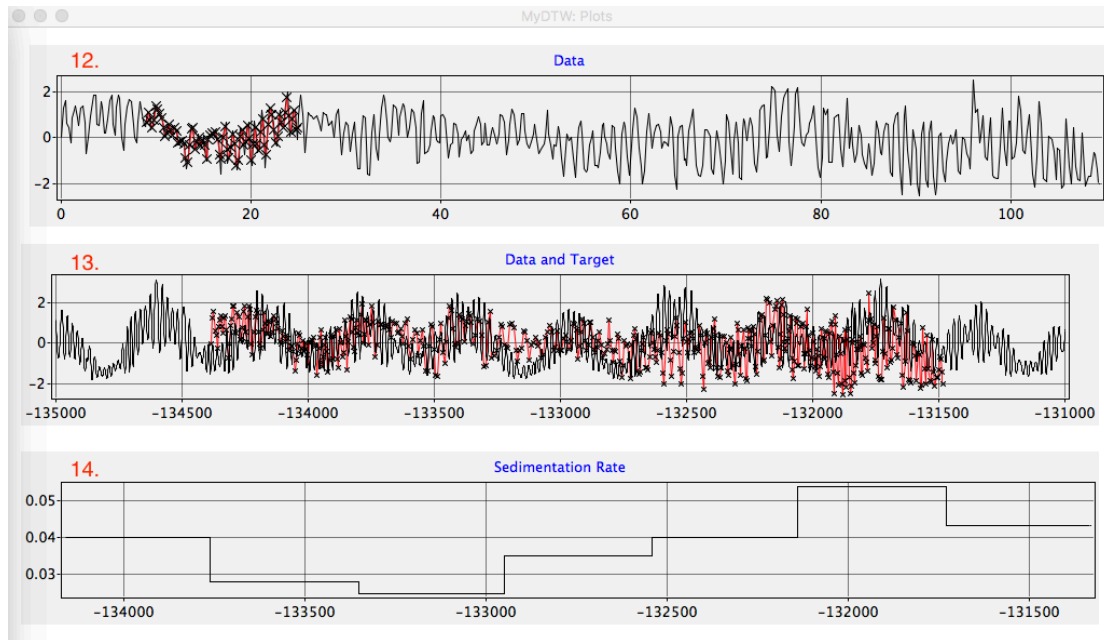


Fig. 8. MyDTW GUI – data pre-processing tab:

- 1 - Open Data File button (text file, two tab or space separated columns [depth/time value]);
- 2 - Open Target File button (text file, two tab or space separated columns [depth/time value]);
- 3 - Open TimeModel File button (text file, two tab or space separated columns [depth/time(data) depth/time(target)]);
- 4 - Select 'Example' Data-Target-TM data set;
- 5 - Select Laskar solution as a target;
- 6 - TimeModel plot;
- 7, 8 - Sub-model selection buttons;
- 9 - Show Full Extent button;
- 10 - Data and Target pre-processing frame;
- 11 - Taner filter frame.
- 12 - Data plot;
- 13 - Target plot;
- 14 - 'Sedimentation rate' plot;

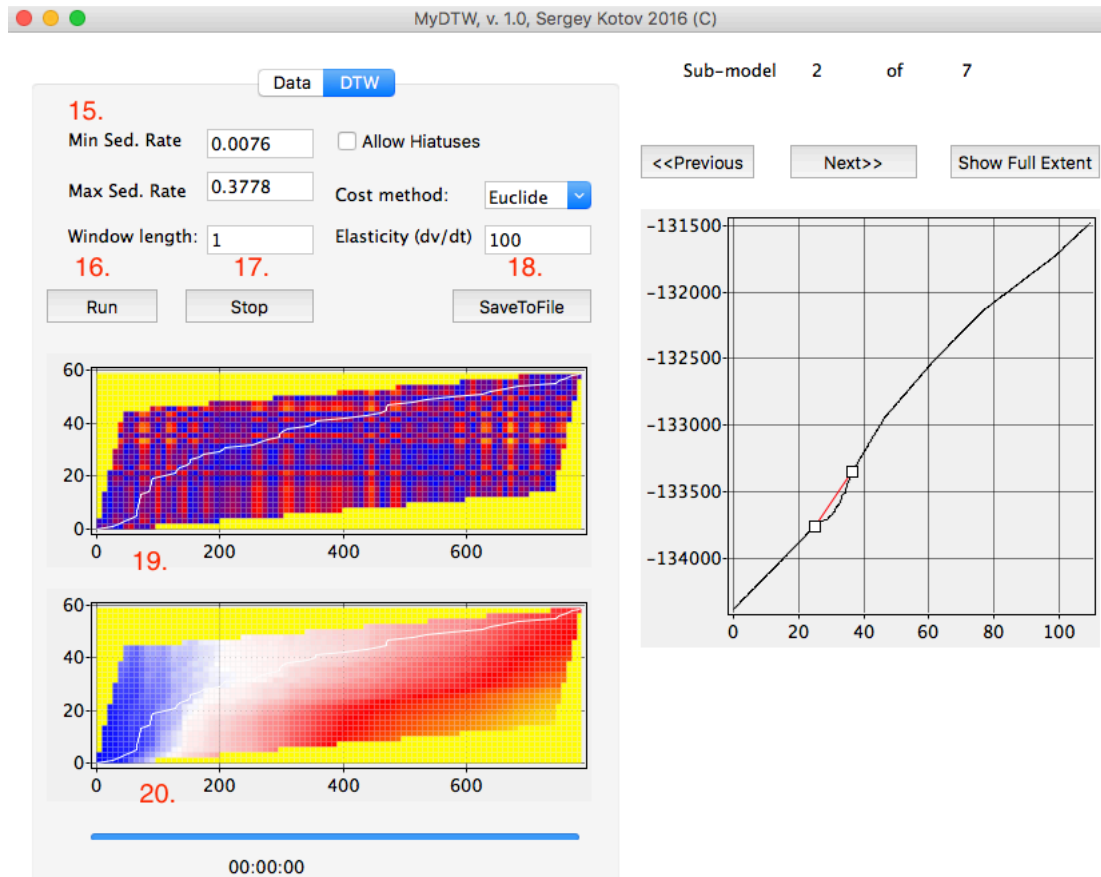


Fig. 9. MyDTW GUI – DTW processing tab:

- 15 – parameters (sedimentation rates etc.);
- 16 – Run button;
- 17 – Stop button;
- 18 – SaveToFile button;
- 19 - cost and accumulate cost matrix plots;
- 20 – progress bar.

Let us pass through main steps of data tuning with MyDTW ($\{n\}$ – anchors in figures 8 and 9).

- 1) Load data, target and rough time model, $\{1, 2, 3\}$. All the data must be arranged as text files with two columns (space, tab, ‘,’ ‘;’ or ‘:’ separated numbers) without header line: [depth/time value] for data and target files, [depth/time(data) depth/time(target)] for tie points of the rough time model. Depths and times should be in increasing order. Time model file must contain at least two tie points. Data and target can be sampled unevenly; they will be automatically resampled evenly with selected steps before the tuning. Alternatively, one can download example data sets in one click $\{4\}$ or select a target from several Laskar astronomical solutions $\{5\}$. Data, target, sedimentation rates, initial time model plots will appear on graphs $\{6-9\}$, all the plots can be zoomed-paned with mouse buttons.

- 2) Pre-process data {13, 14}. Resampling data with different time step will provide different resolution of tuning and processing time (average sampling interval pops up by default). It is also possible to remove linear trend and pre-filter data with Taner's filter (Nyquist frequency pops up by default).
- 3) Select DTW tab. Some 'optimal' parameters will pop up {15}.
- 4) Run tuning {16}. Interrupt tuning if it is too slow {17}, play with different parameters to find appropriate solution. Control results with data and cost matrix plots {6-9, 19, 20}.
- 5) Select next sub-section of the time model {10, 11}. Repeat step 4.
- 6) Save results into the text file {18}. Results are saved as a 4-columns table: ['original depth/time' 'new depth/time' 'normalized data value' 'normalized target value'].

Intermediate results can be saved as a project binary file (main menu – Project – Save as...) and re-opened later for continuation of tuning (main menu – Project – Open). Load data-target-time_model first before saving.

Please, refer the program as [5] when use it in your research.

References:

1. Olea, R.A. Expert systems for automated correlation and interpretation of wireline logs // *Math Geol* (1994) 26: 879. doi:10.1007/BF02083420
2. Lisiecki L. and Lisiecki P. Application of dynamic programming to the correlation of paleoclimate records // *Paleoceanography* (2002), Volume 17, Issue 4, pp. 1-1, CiteID 1049, doi: 10.1029/2001PA000733
3. Pälke, H. Extending the astronomical calibration of the Geological Time Scale PhD thesis, University of Cambridge, (2002)
4. Müller M. *Information Retrieval for Music and Motion* // Springer-Verlag New York, Inc. Secaucus, NJ, USA (2007), ISBN: 3540740473
5. Kotov, S., De Vleeschouwer, D., Martinez, M. and Pälke, H. A signal matching algorithm based on Dynamic Time Warping // *Abstracts, 35IGC, Cape Town, 2016*