TopoZeko: A MATLAB function for 3-D and 4-D topographical visualization in geosciences

Journal Article

Author(s):
Zekollari, Harry

Publication date:
2017-11-23

Permanent link:
https://doi.org/10.3929/ethz-b-000226658

Originally published in:
SoftwareX 6, https://doi.org/10.1016/j.softx.2017.10.004

SoftwareX 6 (2017) 285–292

Contents lists available at ScienceDirect

SoftwareX

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

Original software publication

TopoZeko: A MATLAB function for 3-D and 4-D topographical visualization in geosciences

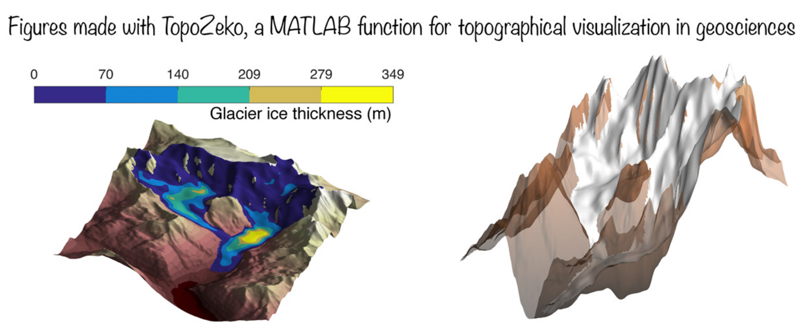
Harry Zekollari *

Laboratory of Hydraulics, Hydrology and Glaciology (VAW), Swiss Federal Institute of Technology (ETH Zürich), Zurich, Switzerland
Swiss Federal Institute for Forest, Snow and Landscape Research (WSL), Birmensdorf, Switzerland
Earth System Science & Departement Geografie, Vrije Universiteit Brussel, Brussel, Belgium

HIGHLIGHTS

- Easy-to-use MATLAB function for 3-D and 4-D geoscientific visualization.
- Visualizations are particularly suited to make time-dependent animations (videos).
- Fast function, that can be used to visualize a variety landscapes at different spatial scales.
- Additional function is included for daily shadowing/insolation cycle visualization.
- Freely available and future updates based on user feedback.

GRAPHICAL ABSTRACT



ARTICLE INFO

Article history:
Received 14 July 2016
Received in revised form 10 September 2017
Accepted 16 October 2017

Keywords:
MATLAB function
Visualization tool
4D visualization
Topography

ABSTRACT

TopoZeko is a MATLAB function for plotting a variety of natural environments with a pronounced topography, such as glaciers, volcanoes and lakes in mountainous regions. This function extends existing MATLAB plotting routines and allows for high-quality 3-D landscape visualization, with a single color defining a featured surface type or with a color scale defining the magnitude of a variable. As an input, only the elevation of the subsurface (typically the bedrock) and the surface are needed, which can be complemented by various input parameters. Several visualization examples are provided alongside with animations, which can directly be generated in the code. Additionally a simple function to calculate the position of the sun is introduced, which can be used to visualize the daily insolation/shadow cycle over a landscape.

Code metadata

Current code version	*v1.0*
Permanent link to code/repository used of this code version	https://github.com/ElsevierSoftwareX/SOFTX-D-16-00059
Legal Code License	*MIT license*
Code versioning system used	*git*
Software code languages, tools, and services used	*MATLAB® 2012a or more recent version*
Compilation requirements, operating environments & dependencies	*Linux, OS X, Microsoft Windows*
If available Link to developer documentation/manual	zekollari.be/topozeko2017
Support email for questions	zharry@ethz.ch, harry.zekollari@wsl.ch, zekollari@gmail.com

* Correspondence to: VAW - ETHZ, Hönggerbergring 26, 8049 Zürich, Switzerland.
E-mail address: zharry@ethz.ch.

http://dx.doi.org/10.1016/j.softx.2017.10.004

1. Motivation and significance

In the geoscientific literature, a variable with a spatial pattern is typically represented in a 2-D plane, where a color scale is used to define its magnitude. This classic visualization method is suited to illustrate the spatial variability of one variable, but it is insufficient to represent both the spatially varying variable and the topography [1]. For this purpose, a 2-D plane can be used in which the two fields (the variable and the topography) are overlapped, but here possibilities are often limited and the illustrations become packed with information (e.g. Fig. 1), potentially leading to a figure that is unclear and unintuitive. Therefore in many cases a 3-D plane representation of the topography is more appropriate (for an elaborate discussion on the advantages and limitations of 2/3/4D visualizations see [2] and [3]). In such a 3-D topographic visualization a color(scale) can be used to represent a variable's spatial cover (e.g. ice, snow, water, lava) (hereafter referred to as a 3-D+ visualization) or spatial variation (e.g. ice thickness, snow depth, water depth, surface velocity, surface gradient) (hereafter referred to as a 4-D visualization). However, in most software used in geo- and environmental sciences, making such 3-D+ and 4-D is not feasible or very complex and time-consuming. This is for instance the case in the widely used numerical computing environment MATLAB.

Here a user-friendly MATLAB function, TopoZeko, is introduced to produce 3-D+ and 4-D visualization of landscapes. Several examples of plots produced by the function are given, as well as examples of time-evolving animations (videos), for which the function is particularly suited. Additionally, a simple function to calculate the position of the sun is presented, which can be used to visualize the daily insolation/shadow cycle over a landscape. TopoZeko belongs to a series of user-friendly tools that have recently been developed to be applicable for a large variety of geoscientific visualization applications, both in MATLAB [5–8] and in other numerical computing environments [9–11].

2. Function description and examples

TopoZeko is based on MATLAB scripts that have been used to visualize the Morteratsch glacier (Switzerland) and the Hans Tausen ice cap (Greenland) in earlier modeling studies [12–15]. These scripts were extended, generalized and transformed into a single MATLAB function in order to be applicable for different settings and purposes. By solely relying on standard MATLAB functions, it is not possible to combine different color schemes within one figure. Through MATLAB's *surf* function a 3-D figure can be generated with one particular color scheme that represents the z-dimension (3rd dimension) or another variable. Distinguishing between two different surface types (e.g. land and water) is not possible in a straightforward way and can only be achieved through different technical steps. TopoZeko deals with these non-trivial technicalities, which make producing 3-D+ and 4-D visualizations in MATLAB a time-consuming and complex task. By means of some technical operations, the correct color bar is displayed and the labeling is adapted accordingly. In brief the following four steps are followed:

(i) The bedrock topography is plotted (and eventually its transparency is adapted)
(ii) The color scheme (*colormap* in MATLAB) of the figure is transformed into a concatenated color scheme consisting of the surface color scheme (one color for 3-D+ plots, a range of colors for 4-D plots) and the bedrock color scheme
(iii) The limits of the color scheme are adapted (through *caxis*)
(iv) The surface topography is plotted at its real elevation (and eventually its transparency) is adapted and it is colored in order to fulfill the function input

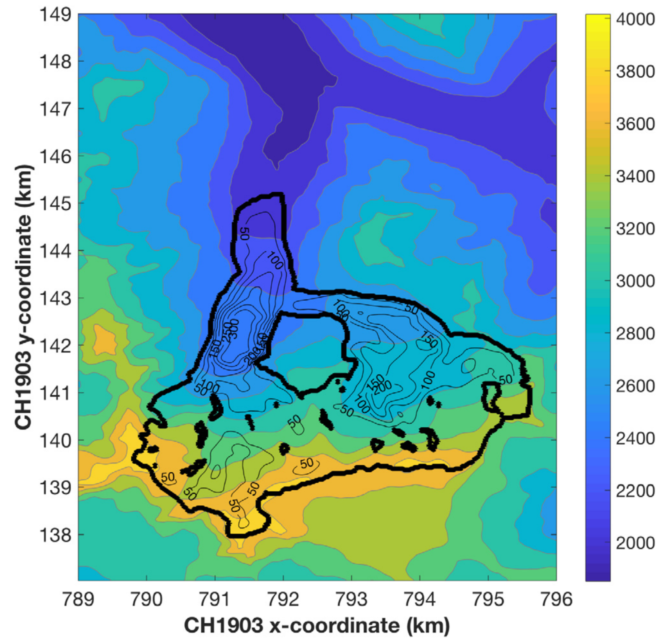


Fig. 1. 2-D visualization of the Morteratsch glacier and surrounding topography (elevation in m a.s.l.) in 2001. The thick black line represents the glacier outline, while the thin black lines are the ice thickness contours (50 m interval) (data from [4]). This figure was created using standard MATLAB functions *pcolor*, *contourf* and *contour*.

TopoZeko thus largely extends the standard 3-D MATLAB plotting options and offers new visualization possibilities that can be of great use for geoscientific applications. Other MATLAB topography plotting toolboxes exist (e.g. TopoToolbox [5]) but these do not allow for 3-D color visualizations that combine different color scales.

The only two mandatory function inputs are the subsurface elevation (typically the bedrock) (first input) and the surface elevation (second input), which need to be given as 2-D matrices of equal size. In the most basic case the function call reads:

TopoZeko(bedrock_elevation_matrix,surface_elevation_matrix)

This can be complemented with up to 40 optional input parameters in order to adapt the figure to specific needs (Table 1) (see Appendix A for examples).

For the following examples, Digital Elevation Models (DEMs) from the Galápagos Islands (Ecuador) (100 m resolution, covering an area of 19 600 km^2) and the Morteratsch glacier (Switzerland) (25 m resolution, covering an area of 84 km^2) and its reconstructed bedrock topography [4] are used.

2.1. 3-D+ visualization

3-D+ visualizations are generated when the optional parameter *extra_dimension* is not defined (see also Table 1). For all grid points where the subsurface (bedrock) elevation differs from the surface elevation, the surface is displayed in the color defined by *sur_color* (when *sur_color* is not defined, white is used as a standard). In Fig. 2 several applications of TopoZeko for the Morteratsch glacier are shown. Fig. 2a is a standard figure (*view_orientation* is the only optional parameter that is defined, see Appendix A for function calls), in which the ice surface is white. Fig. 2b illustrates the same environment, but with a different point of view and light source position, an inversed color scale for the bedrock and additionally the axes are displayed. In Fig. 2c the glacier is visualized from a lateral point of view and the bedrock is transparent (50% transparency), which shows the pronounced bedrock overdeepening

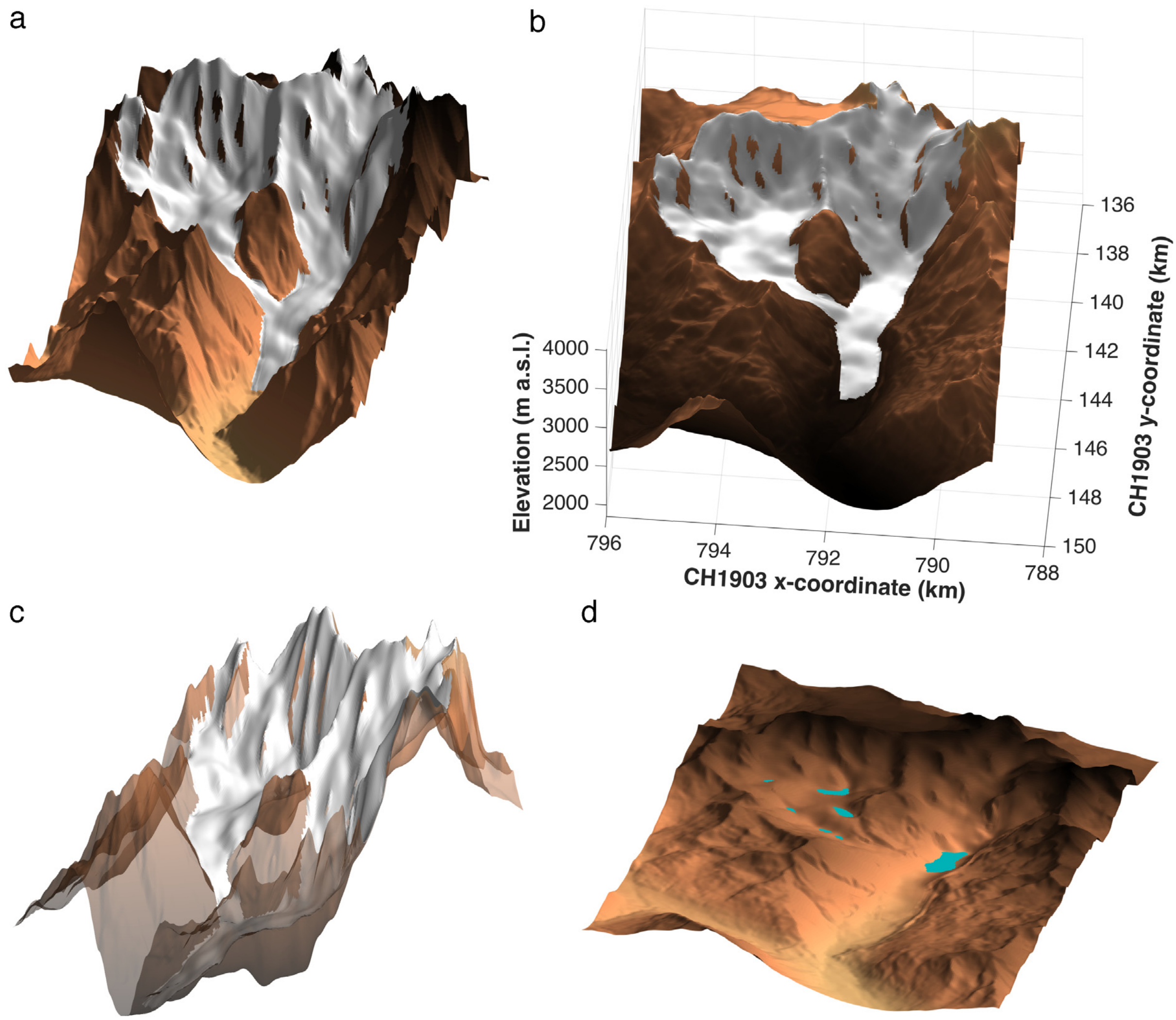


Fig. 2. 3-D+ visualizations of the Morteratsch glacier obtained by adapting the input variables of TopoZeko. Panel a, b and c represent the situation in 2001, while panel d is a possible future landscape without ice, where all the bedrock depressions are filled with water. See Appendix A for function calls.

around the convergence area of the glacier. In the case of a glacier retreat, these kind of overdeepenings could become glacial lakes, as is illustrated in Fig. 2d, where the vertical dimension is downscaled, all ice is removed and the bedrock depressions are filled with water. The axes in Fig. 2 are not equidistant, but if required this can be obtained by modifying the range of the axes (optional parameters *xlim*, *ylim* and *zlim*) and/or by changing the size of the figure (optional parameter *size_cm*, *size_pix*).

TopoZeko can also be applied for very different landscapes and on a totally different scale. This is illustrated in Fig. 3 for the two westernmost islands of the Galápagos archipelago (Ecuador): Isla Isabela (the largest island) and the adjacent Isla Fernandina (Fig. 2 covers 84 km^2, while Fig. 3 covers 19 600 km^2). Here the surface of the calderas is shown in red, which is a hypothetical landscape with active surface volcanism.

2.2. 4-D visualization

4-D visualizations are generated when the variable *extra_dimension* is defined. When *extra_dimension* is defined as *on* (e.g.: *TopoZeko(bedrock_elevation_matrix,surface_elevation_matrix, 'extra_dimension','on')*) the difference between the surface and the subsurface (e.g. ice thickness, lake depth, ...) is plotted as a variable. Alternatively another (external) variable (e.g. ice thickness change, surface velocity, surface gradient, ...) can be used as an input for extra_dimension (e.g.: *TopoZeko(bedrock_elevation_matrix,surface_elevation_matrix, 'extra_dimension',surface_velocity)*). Fig. 4a is similar to the classic visualization of the glacier (Fig. 2a), but here the ice thickness (the difference between the surface and the bedrock elevation), which varies between 0 and 349 m, is represented as an extra dimension (cf. 2-D representation in Fig. 1). Fig. 4b shows how the ice thickness can easily be represented in a very different way, with another vertical scaling, a different view orientation, and other (discrete) color schemes for the bedrock and the 4th dimension (i.e. the ice thickness). The part of the glacier with the highest ice thickness corresponds to a zone with a bedrock overdeepening (see also Fig. 2c). In case all ice would disappear and this depression would fill up with water, the resulting lake would be up to 90 m deep here, as is illustrated in Fig. 4c.

3. Animations

The figures made by TopoZeko are particularly suited to make animations by image sequencing. The animations can be made

Table 1
Overview of optional parameters when for the TopoZeko function. Note that American spelling (e.g. 'bed_colors'), which is standard in MATLAB, is used for the variable names.

Parameter	3-D+	4-D	Description	Standard value	(other) possible values and examples
axes	✓	✓	Show the axes	*'off'*	*'on'*
bed_colors	✓	✓	Number of colors for bedrock	128	Integer, should be larger than 1
bed_colormap	✓	✓	MATLAB colormap (color scheme) for the bedrock	*'copper'*	All standard MATLAB colormaps or a user-defined colormap
bed_colormap_flipud	✓	✓	Inverse the colormap (color scheme) for the bedrock	*'on'*	*'off'*
bed_trans	✓	✓	Transparency of the bedrock	1	Real number between 0 and 1
caxis		✓	Range of the color bar		[*c1 c2*] where *c1* is smaller than *c2* and both are real numbers
cbar_colors		✓	Number of colors for the 4th dimension	128	Integer, should be larger than 1
cbar_position		✓	Position of the color bar	*'northoutside'*	MATLAB positioning (e.g. *'north'*)
cbar_tick_format		✓	Format for the ticks on the color bar		e.g.1. *'%d4.2'* (double, field width 4, precision 2) e.g.2. *5*: 5 relevant digits
D2	✓	✓	Make an additional 2-D plot of the 4th dimension (thickness/depth, extra variable)	*'off'*	*'on'*
D4_colormap		✓	MATLAB colormap (color scheme) for the 4th dimension	*'jet'*	All standard MATLAB colormaps or a user-defined colormap
D4_colormap_flipud		✓	Inverse the colormap (color scheme) for the 4th dimension	*'off'*	*'on'*
extra_dimension		✓	Whether or not to plot a 4th dimension		*'on'*: 4th dimension is the thickness/depth field *'variable'*: variable will be plotted as the 4th dimension
label_size	✓	✓	Font size of all labels (on axes, color bars, for x, y, z-labels and for title). When defined, all other optional label size parameters are neglected		Positive integer
light_orientation	✓	✓	Orientation of the light source	[−90 45]	[*lo1 lo2*] where *lo1* is the azimuth and *lo2* the height (given as an angle)
size_cm	✓	✓	Image size of the plot in cm	[20 20]	[*s1 s2*] where *s1* and *s2* are positive real numbers
size_pix	✓	✓	Image size of the plot in pixels		[*p1 p2*] where *p1* and *p2* are positive real numbers
sur_color	✓		Color of the ice/lava/lake/etc. surface in 3-D plot	[1 1 1] (white)	[1 0 0] (red) [0 1 0] (green) [0 0 1] (blue) [1 1 0] (yellow) …
sur_material	✓	✓	Appearance of surface material	*'dull'*	*'shiny'*, *'metal'*
sur_trans	✓	✓	Transparency of the surface	1	Real number between 0 and 1
tick	✓	✓	Show the ticks on the axes	*'on'*	*'off'*
tick_size	✓	✓	Font size of all axes and color bars ticks	18	Positive integer
title	✓	✓	Title of the figure		e.g. 'Morteratsch glacier'
title_size	✓	✓	Font size of the title	22	Positive integer
vertical_scaling	✓	✓	Fraction of the z-axis that is used to depict the topography	1	Real number between 0 and 1
view_orientation	✓	✓	Orientation of camera view point	[0 45]	[*vo1 vo2*] where *vo1* is the azimuth and *vo2* the height (given as an angle)
xlabel	✓	✓	Label on the x-axis		e.g. 'this is the x-label'
xlabel_rotation	✓	✓	Rotation of the x-label	0	Real number
xlabel_size	✓	✓	Font size of the x-label	18	Positive integer
xlim	✓	✓	Range of the x-axis		[*x1 x2*] where *x1* is smaller than *x2* and both are real numbers
xvalues	✓	✓	First and last column x-values		[*xval1 xval2*] where *xval1* is smaller than *xval2* and both are real numbers
ylabel	✓	✓	Label on the y-axis		e.g. 'this is the y-label'
ylabel_rotation	✓	✓	Rotation of the y-label	0	Real number
ylabel_size	✓	✓	Font size of the y-label	18	Positive integer
ylim	✓	✓	Range of the y-axis		[*y1 y2*] where *y1* is smaller than *y2* and both are real numbers
yvalues	✓	✓	First and last rows y-values		[*yval1 yval2*] where *yval1* is smaller than *yval2* and both are real numbers
zlabel	✓	✓	Label on the z-axis		e.g. 'this is the z-label'
zlabel_rotation	✓	✓	Rotation of the z-label	90	Real number
zlabel_size	✓	✓	Font size of the z-label	18	Positive integer
zlim	✓	✓	Range of the z-axis		[*z1 z2*] where *z1* is smaller than *z2* and both are real numbers

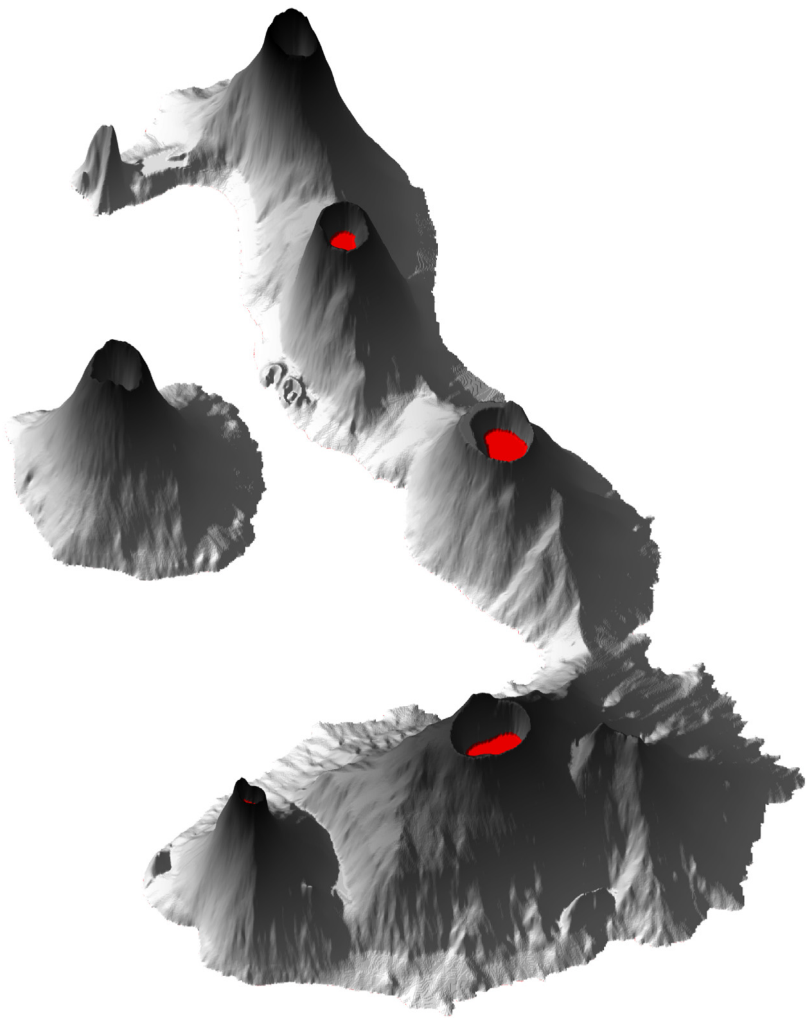


Fig. 3. 3-D+ visualization of Isla Isabela (left) and Isla Fernandina (right) (Galápagos Islands, Ecuador) obtained by adapting the input variables of TopoZeko. The surface of the caldera is shown in red. See Appendix A for function calls. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

directly in MATLAB (see the provided scripts for more information), but the size of the video is limited to the screen size in this case. In case a higher quality is needed it is recommended to save all visualizations as images (examples in the provided scripts) and subsequently make an animation of this in a dedicated program such as Photoshop, GIMP (General Image Manipulation Program) or QuickTime Pro.

3.1. Retreat of the Morteratsch glacier

A simple parameterization for the retreat of the Morteratsch glacier in a warming climate is developed based on field observations. It is assumed that the glacier merely changes (surface elevation change of -0.1 m a^{-1}) in the accumulation area, i.e. above the Equilibrium Line Altitude (ELA), which is around 3000 m in 2001 (year of the DEM). In the ablation area (zone under the ELA) the ice thickness change increases with lowering elevation, which agrees with observations of Alpine glaciers [16,17]. Here the surface elevation change gradient is set to -0.005 m m^{-1}, a value that agrees well with local field observations and that yields realistic results for the period 2001–2015. With this parameterization: (i) the glacier front retreats by 425 m for this period (close to the observed 470 m retreat), and (ii) the main tributary glacier disconnects in 2014 (which occurred in 2015 in reality). This parameterization is fixed in time and only the ELA increases, which is set to a rate of + 6 m a^{-1}. This broadly corresponds to an extreme warming scenario of +4 °C over a century, assuming an annual temperature lapse rate of -0.007 °C m^{-1}. A 3-D+ (**video 1**) and a 4-D animation representing the magnitude of the annual surface elevation change (i.e. annual ice thickness reduction) (**video 2**) were created based on this parameterization for the period 2001–2100. Snapshots of these animations are shown in Fig. 5.

3.2. Daily insolation over Isla Fernandina and Isla Isabela

In a second example an animation is made of the daily insolation/shadow cycle over Isla Fernandina and Isla Isabela (Galápagos Islands). This is done for an equidistant *xy*-plane, but the vertical dimension is exaggerated in order to have more pronounced shadows for the animation. The position of the sun throughout the day is calculated through a simple function (SunZeko) that requires three inputs: the declination of the sun ($-23.44°$ at winter solstice, 0° at vernal and autumnal equinox, $+23.44°$ at summer solstice), the geographic latitude and the local hour angle. Some simplifications that have a minor effect are: (i) there is no time-correction (i.e. sun culminates exactly at noon), (ii) there is no refraction of

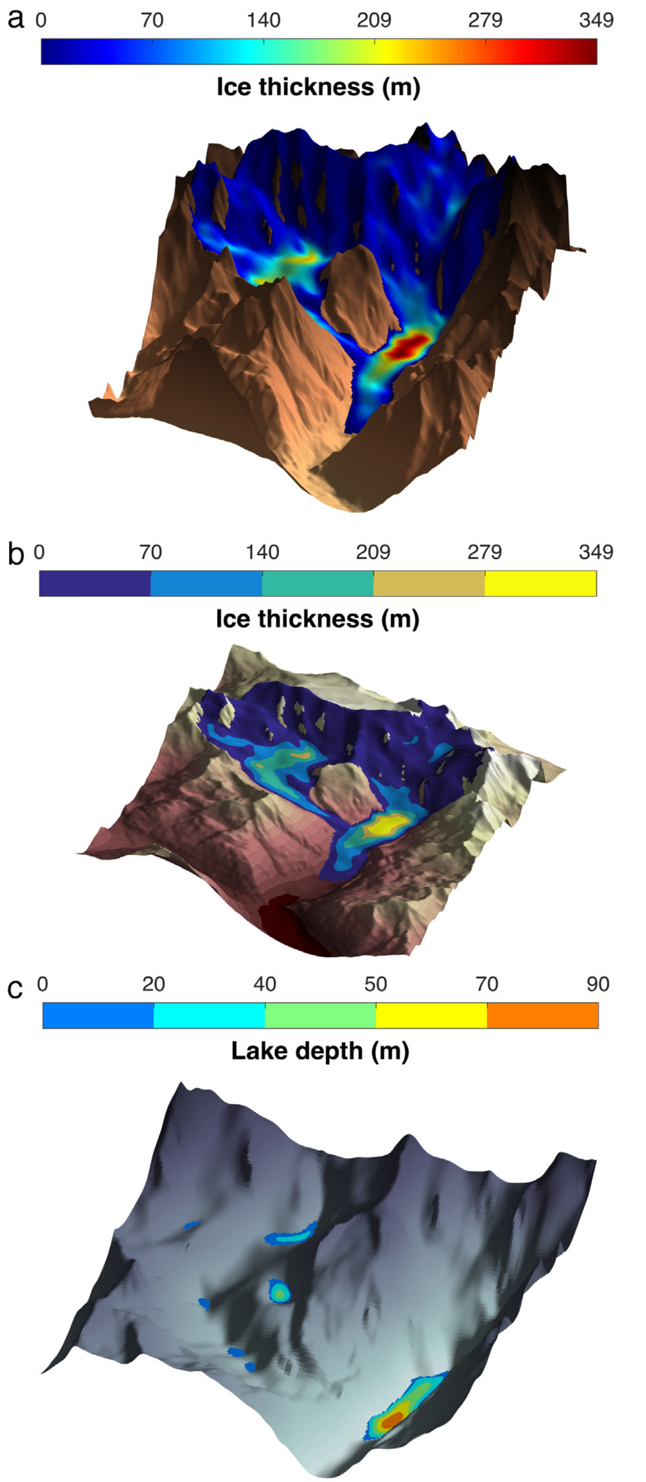


Fig. 4. 4-D visualization of the Morteratsch glacier obtained by adapting the input variables of TopoZeko. (a&b) Glacier geometry and ice thickness in 2001 obtained from different input parameters. (c) lake depth when all the bedrock depressions are water-filled (cf. Fig. 2d). See Appendix A for function calls.

light and (iii) the sun is considered as a point source. Here, as an example the situation at the summer solstice (declination of the sun is $+23.44°$) is considered, in which the sun rises around the ENE, culminates in the N (sun height is 66.06°) and sets around the WNW. The position of the sun is set as the position of the light source (through the optional parameter *light_orientation*, see Table 1) to make an animation of the daily insolation/shadow cycle (**video 3**). In Fig. 6 snapshots of the resulting animation are shown, where the orientation of the shadow changes throughout the day. This kind of animation can be used to easily visualize the daily insolation by natural features and are strong and easy-to-use tools for educational and research purposes.

4. System requirements and user recommendations

The visualizations can be modified to the specific needs of the user by using optional input parameters (Table 1), but additionally the code is documented with many comments, which makes it feasible to adapt certain features. Only the basic package of MATLAB is needed and the code can be run on any standard stationary or portable personal computer. Some recently introduced MATLAB features were avoided in order to make the code compatible with older versions (this is documented in the code). The code was tested on several versions of MATLAB (From R2012a tot R2017a, both on Windows and Mac) and the function was very fast for the Morteratsch glacier (in the order of 0.1 s for 3-D+ visualizations, around 0.3 s for 4-D visualizations) (matrix of $480 \times 281 = 134\,880$ grid points) and for the Galápagos islands (in the order of 0.5 s for 3-D+ visualizations) (matrix of $1400 \times 1400 = 1.96$ million grid points) on a standard modern laptop (Fig. 7).

When working with high-resolution data or files of very large sizes, visualization of the original data may require more time and/or memory to be generated (Fig. 7). In such cases the detail of data input (original resolution) will however not be visually detectable. For such applications it is therefore advised to first upscale the data to a lower resolution (e.g. through MATLAB's *imresize* function) or in case the original data resolution is of importance, only visualize the region of interest. If the upscaling occurs to a resolution of a million pixels, the upscaled data visualization will be rapid (a few seconds at most, see Fig. 7) and visually indistinguishable from the original data. For LIDAR data it is advised to first apply a rasterization/interpolation to a regular grid (and eventually lowering its resolution), after which the data can be visualized with TopoZeko.

5. Impact and conclusions

TopoZeko is an easy to use function that allows creating 3-D+ and 4-D visualizations and animations in MATLAB. This tool extends the existing visualization options in MATLAB and is particularly suited for settings with a pronounced topography, which are often prominent in geosciences and related research fields. Scientists working in the numerical environment MATLAB and wishing to produce sophisticated plots of landscapes within this environment will not be confronted to the intrinsic technicalities and limitations related to using different color schemes within a single visualization. Besides the mandatory input (bedrock and surface elevation), a large variety of optional input parameters can be specified to adapt the figures to the specific needs. Users are invited to use TopoZeko and to provide feedback that will be used to extend and improve this function in future versions.

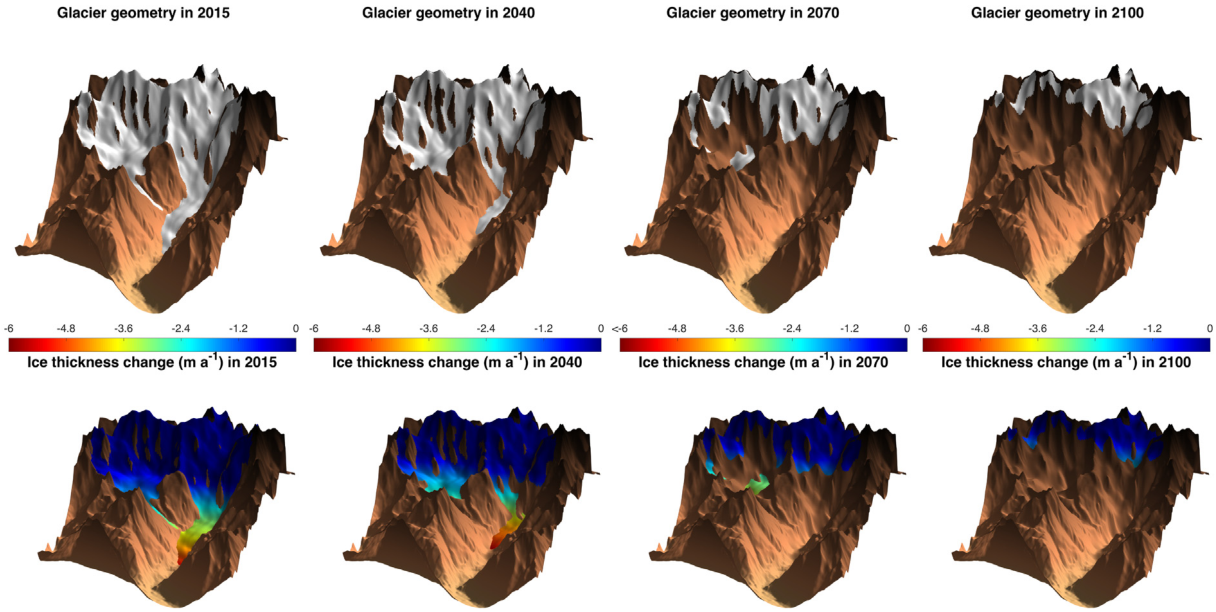


Fig. 5. Snapshots of the 3-D+ (upper row) and the 4-D animations (lower row) of the retreat of the Morteratsch glacier in a warming climate (video 1 and video 2).

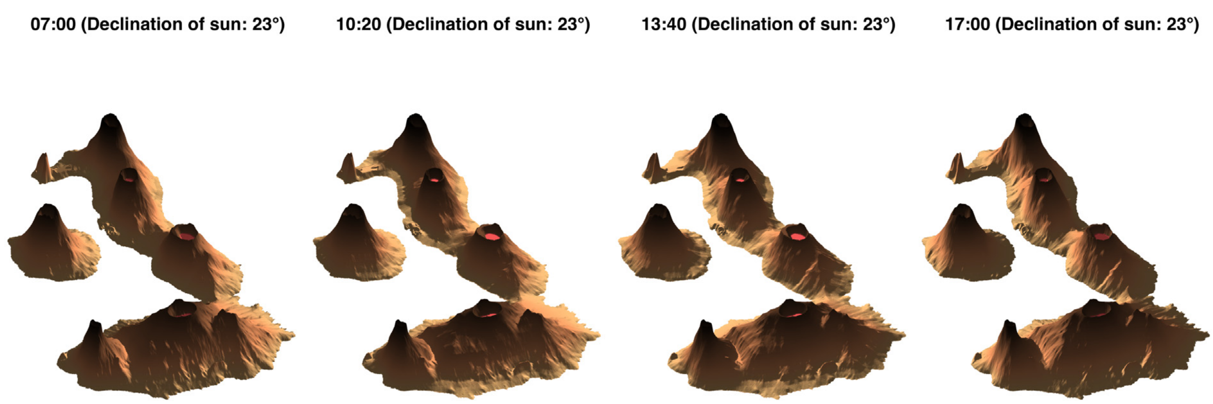


Fig. 6. Animation of daily insolation/shadow cycle over Isla Isabela and Isla Fernandina (Galápagos Islands, Ecuador) (video 3).

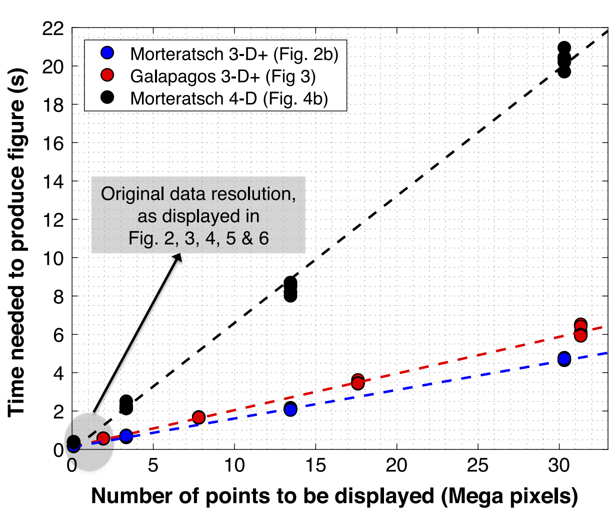


Fig. 7. Time needed to generate various figures with a modern laptop (2,2 GHz Intel Core i7; 16 GB 1600 MHz DDR3; Intel Iris Pro 1536 MB integrated graphic card) in MATLAB 2017a. The high-resolution datasets were obtained through a nearest-neighbor downscaling of the original data. The original data (as displayed in Figs. 2–6) is in the gray circle.

Acknowledgments

Four anonymous reviewers are thanked for their comments that helped to improve the paper. The DEM from the Galápagos Islands was freely available and downloaded from http://www.rsgis.ait.ac.th/~souris/ecuador.htm. Harry Zekollari realized a part of this work through a Ph.D. fellowship of the Research Foundation-Flanders (FWO-Vlaanderen). Sofie Brouwers, Wim Thiery, David De Vleeschouwer and Tom Van der Stocken are thanked for the encouraging discussions.

Appendix A. Function call for figures

Figure 2a: *TopoZeko(BED, SUR, 'view_orientation',[-159.5 40]);*

Figure 2b: *TopoZeko(BED, SUR ,'axes','on', 'bed_colormap_flipud','off', 'label_size',18, 'light_orientation',[-90 90], 'view_orientation',[-175 64], 'xlabel','CH1903 x-coordinate (km)', 'xlabel_rotation',-3, 'xvalues',[789 796], 'ylabel','CH1903 y-coordinate (km)', 'ylabel_rotation',85, 'yvalues',[137.025 149], 'zlabel','Elevation (m a.s.l.)');*

Figure 2c: *TopoZeko(BED, SUR, 'bed_colormap_flipud','off', 'bed_trans',0.5, 'view_orientation',[-113 15], 'xlim',[50 280], 'ylim',[0 350]);*

Figure 2d: *TopoZeko(BED, SUR_LAKE, 'sur_color',[0 206/255 209/255], 'vertical_scaling',0.25, 'view_orientation',[-159.5 40]);*

Figure 3: *TopoZeko(BED, SUR, 'bed_colormap','gray', 'sur_color',[1 0 0], 'vertical_scaling',0.7, 'view_orientation',[0 80], 'xlim',[0 1400], 'ylim',[0 1400]);*

Figure 4a: *TopoZeko(BED, SUR, 'extra_dimension','on', 'view_orientation',[-159.5 40]);*

Figure 4b: *TopoZeko(BED,SUR, 'bed_colors',20, 'bed_colormap',' pink', ,'bed_colormap_flipud','off', 'cbar_colors',5, 'D4_colormap',' parula', 'extra_dimension','on', 'vertical_scaling',0.5, 'view_orientation',[-151 53]);*

Figure 4c: *TopoZeko(BED, SUR_LAKE, 'bed_colormap','bone', 'cbar_colors',5, 'extra_dimension','on', 'view_orientation',[-159.5 40], 'xlim',[50 250], 'ylim',[130 260]);*

Appendix B. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.softx.2017.10.004.

References

[1] MacEachren AM, Taylor DR. Visualization in modern cartography. 2nd ed. Oxford: Pergamon; 1994.

[2] Nielson G, Hagen H, Müller H. Scientific visualization: Overviews, methodologies and techniques. IEEE Computer Society Press; 1997.

[3] Wilkinson L. The grammar of graphics. 2nd ed. New York: Springer; 2005. http://dx.doi.org/10.1007/0-387-28695-0.

[4] Zekollari H, Huybrechts P, Fürst JJ, Rybak O, Eisen O. Calibration of a higher-order 3-D ice-flow model of the Morteratsch glacier complex, Engadin, Switzerland. Ann Glaciol 2013;54:343–51. http://dx.doi.org/10.3189/2013AoG63A434.

[5] Schwanghart W, Kuhn NJ. TopoToolbox: A set of Matlab functions for topographic analysis. Environ Model Softw 2010;25:770–81. http://dx.doi.org/10.1016/j.envsoft.2009.12.002.

[6] Bezděk A, Sebera J. Matlab script for 3D visualizing geodata on a rotating globe. Comput Geosci 2013;56:127–30. http://dx.doi.org/10.1016/j.cageo.2013.03.007.

[7] Pianosi F, Sarrazin F, Wagener T. A matlab toolbox for global sensitivity analysis. Environ Model Softw 2015;70:80–5. http://dx.doi.org/10.1016/j.envsoft.2015.04.009.

[8] Wu L, Kravchinsky VA, Potter DK. PMTec: A new MATLAB toolbox for absolute plate motion reconstructions from paleomagnetism. Comput Geosci 2015;82:139–51. http://dx.doi.org/10.1016/j.cageo.2015.06.009.

[9] Horsburgh JS, Reeder SL, Jones AS, Meline J. Open source software for visualization and quality control of continuous hydrologic and water quality sensor data. Environ Model Softw 2015;70:32–44. http://dx.doi.org/10.1016/j.envsoft.2015.04.002.

[10] Paciello R, Coviello I, Bitonto P, Donvito A, Filizzola C, Genzano N, Lisi M, Pergola N, Sileo G, Tramutoli V. An innovative system for sharing, integration and visualization of heterogeneous 4D-information. Environ Model Softw 2016;77:50–62. http://dx.doi.org/10.1016/j.envsoft.2015.11.011.

[11] Zhang T, Li J, Liu Q, Huang Q. A cloud-enabled remote visualization tool for time-varying climate data analytics. Environ Model Softw 2016;75:513–8. http://dx.doi.org/10.1016/j.envsoft.2015.10.033.

[12] Zekollari H, Fürst JJ, Huybrechts P. Modelling the evolution of Vadret da Morteratsch, Switzerland, since the Little Ice Age and into the future. J Glaciol 2014;60:1208–20. http://dx.doi.org/10.3189/2014JoG14J053.

[13] Zekollari H, Huybrechts P, Noël B, van de Berg WJ, van den Broeke MR. Sensitivity, stability and future evolution of the world's northernmost ice cap, Hans Tausen Iskappe (Greenland). Cryosph 2017;11:805–25. http://dx.doi.org/10.5194/tc-11-805-2017.

[14] Zekollari H, Lecavalier BS, Huybrechts P. Holocene evolution of Hans Tausen Iskappe (Greenland) and implications for the palaeoclimatic evolution of the high Arctic. Quat Sci Rev 2017;168:182–93. http://dx.doi.org/10.1016/j.quascirev.2017.05.010.

[15] Zekollari H, Huybrechts P. On the climate-geometry imbalance, response time and volume-area scaling of an alpine glacier: insights from a 3-D flow model applied to Vadret da Morteratsch, Switzerland. Ann Glaciol 2015;56:51–62. http://dx.doi.org/10.3189/2015AoG70A921.

[16] Huss M, Jouvet G, Farinotti D, Bauder A. Future high-mountain hydrology: A new parameterization of glacier retreat. Hydrol Earth Syst Sci 2010;14:815–29. http://dx.doi.org/10.5194/hess-14-815-2010.

[17] Fischer M, Huss M, Hoelzle M. Surface elevation and mass changes of all swiss glaciers 1980-2010. Cryosph 2015;9:525–40. http://dx.doi.org/10.5194/tc-9-525-2015.