



Short Communication: TopoToolbox 2 – MATLAB-based software for topographic analysis and modeling in Earth surface sciences

W. Schwanghart^{1,2} and D. Scherler³

¹Geohazards, University of Potsdam, Institute of Earth and Environmental Science, Karl-Liebknecht-Str. 24–25, 14476 Potsdam-Golm, Germany

²Department of Agroecology, Aarhus University, Blichers Allé 20, 8830 Tjele, Denmark

³Geological and Planetary Sciences, California Institute of Technology, 1200 E California Blvd, Mailcode 100-23, Pasadena, CA 91125, USA

Correspondence to: W. Schwanghart (w.schwanghart@geo.uni-potsdam.de)

Received: 22 July 2013 – Published in Earth Surf. Dynam. Discuss.: 16 August 2013

Revised: 13 December 2013 – Accepted: 27 December 2013 – Published: 15 January 2014

Abstract. TopoToolbox is a MATLAB program for the analysis of digital elevation models (DEMs). With the release of version 2, the software adopts an object-oriented programming (OOP) approach to work with gridded DEMs and derived data such as flow directions and stream networks. The introduction of a novel technique to store flow directions as topologically ordered vectors of indices enables calculation of flow-related attributes such as flow accumulation ~20 times faster than conventional algorithms while at the same time reducing memory overhead to 33 % of that required by the previous version. Graphical user interfaces (GUIs) enable visual exploration and interaction with DEMs and derivatives and provide access to tools targeted at fluvial and tectonic geomorphologists. With its new release, TopoToolbox has become a more memory-efficient and faster tool for basic and advanced digital terrain analysis that can be used as a framework for building hydrological and geomorphological models in MATLAB.

1 Introduction

An increasing number of research studies use digital elevation models (DEMs) for automated spatial analysis and advanced process-based modeling. Although geographical information system (GIS) software packages such as ESRI's ArcGIS include various interfaces for implementing user-specific codes and models, many users prefer programming environments such as MATLAB that already include large libraries for different computational tasks. Since its release in 2010, TopoToolbox, a code library for MATLAB, has been used in various studies of the Earth's surface and surficial water and material fluxes. Applications include topics such as supraglacial meltwater dynamics (Clason et al., 2012), contaminant transport in streams and groundwater (Messier et al., 2012), tectonic geomorphology (Shahzad and Gloaguen, 2011; Scherler et al., 2013), soil nutrient dynam-

ics (Schwanghart and Jarmer, 2011), karst hydrology (Borghi et al., 2011) and flood modeling (de Moel et al., 2012).

Despite broad attention and positive feedback, the lack of a user-friendly graphical user interface (GUI) may also have discouraged interested users who favor a visual approach towards DEM processing and analysis. Moreover, processing very large DEMs, or, within a modeling framework, processing certain computations a large number of times, is a common challenge in modern geomorphological studies and requires efficient computational techniques. While very large DEMs are most easily handled with software that implements efficient input/output (I/O) communication between internal and external memory, a strategy pursued by ArcGIS, GRASS (Jasiewicz, 2011) and more specialized DEM processing software such as terraflo (Arge et al., 2003), the memory constraint exerted by MATLAB's primary use of the main memory is an obstacle even for medium-sized

DEMs. Although the spread of 64 bit computers has allayed the memory problem somewhat, MATLAB's heavy use of double-precision data imposes limits that are too rigid for larger analyses and modeling tasks.

Here we introduce several important modifications and various extensions that come with the release of version 2 of the TopoToolbox. Specifically, we report the development of an object-oriented design towards the representation of grids, flow routing, and stream networks and give account of a suite of GUIs that provide easy access to a variety of new and existing methods frequently used in topographic studies. Our strategy to reduce memory constraints, which have been particularly pertinent to flow-related algorithms, enables much faster and memory-efficient work flow and thus the processing of much larger DEMs than with previous versions of TopoToolbox. We believe that these advances will promote the development of tools and models in Earth surface sciences.

2 Object-oriented design

Gridded DEMs and terrain attributes incorporate a variety of information other than a gridded set of points. A DEM has physical units and is georeferenced; that is, it is associated with physical space by a set of geographical or map coordinates. In addition, DEM derivatives (or terrain attributes) are tightly linked to algorithms and models and lend themselves to specific types of analysis (Pike et al., 2009). These task- and object-specific properties provide the basis for an object-oriented approach to DEMs. In object-oriented programming (OOP), data, attributes, and procedures are coupled into objects that encapsulate the model of what each object represents (Register, 2007). Typical objects in DEM analysis are surface flow paths and the resulting stream networks. In TopoToolbox the aim of using OOP is to organize data and functions into classes with specific object-related attributes that organize and encapsulate information on spatial referencing, flow topology, and geometry and thereby facilitate data handling and analysis.

TopoToolbox 2 includes three main object classes: GRIDObj, a class to store and analyze gridded data (grid objects); FLOWObj, a class for flow-path networks (flow objects) that is derived from an instance of GRIDObj; and STREAMObj, which represents the channelized fraction (stream objects) of the flow-path network from which it is derived. Additional object classes, such as SWATHObj, which contains swath profiles that are derived along directed linear features (e.g., an instance of STREAMObj), are more tool-specific and not as tightly coupled within TopoToolbox as the three main object classes. These classes are templates for the creation of specific instances of each class. Whereas an instance of GRIDObj simply contains the data grid as well as georeferencing data, instances of FLOWObj and STREAMObj implement a novel storage approach that

represents the directed network of flow paths and stream networks, respectively.

3 A graph-theoretical representation of flow directions

Topography governs the transport mode and the magnitude of lateral material and water fluxes on the Earth's surface and in the near-surface underground. Among topographic attributes, the flow direction probably belongs to the most frequently used, because it forms the basis for many other hydrology-related variables that explain or control many key environmental processes (Wilson and Gallant, 2000). Algorithms to derive flow directions are classified into single-neighbor and multiple-neighbor flow algorithms (Gruber and Peckham, 2009). Single-neighbor algorithms pass the flow from each cell to its lowest neighboring cell (usually referred to as the SFD or D8 algorithm; O'Callaghan and Mark, 1984). Multiple-neighbor algorithms, such as MFD (Quinn et al., 1991), D_∞ (Tarboton, 1997), DEMON (Costa-Cabral and Burges, 1994) or the mass-flux method (Gruber and Peckham, 2009), encompass a number of different approaches to model dispersive flow, which is particularly important for simulating water distribution on hillslopes.

TopoToolbox supports the SFD and MFD algorithms. In contrast to other GIS software, in the first version of TopoToolbox, flow directions were stored in a sparse, weighted adjacency matrix to represent the flow network. While this strategy allows for calculation of flow accumulation and other flow-related terrain attributes using MATLAB's built-in sparse matrix routines (Schwanghart and Kuhn, 2010), it also requires large memory space. The MFD and SFD matrix require approximately six and two times the memory space of the DEM, respectively, if all variables are stored in the same data formats.

In the newly introduced flow direction class (FLOWObj), the storage of flow directions takes advantage of the fact that flow networks can be modeled as directed acyclic graphs (DAGs). Our approach is similar to MATLAB's sparse matrix storage organization but uses the "tuple" format, which is a collection of row, column and value 3-tuples of the nonzero elements of the flow direction matrix (Kepner, 2011). The important difference is that row and column indices are stored in topologically descending order, an approach that was recently adopted in a numerical landscape evolution model (LEM) (Braun and Willett, 2012) and a dynamic overland flow-routing model (Huang and Lee, 2013). Topological sorting is only possible for DAGs and refers to the permutation of nodes that brings the flow direction matrix to a lower triangular form (Chen and Jacquemin, 1988). The storage of flow directions in topological order in a FLOWObj speeds up the computation of flow-related topographic attributes by about 500 % compared to the previous version of TopoToolbox. Furthermore, the FLOWObj requires 89 %

Table 1. Performance comparison between TopoToolbox 2 and similar software. Performance is evaluated as computation time in seconds required to run a specific tool on a computer with an Intel Core i5 CPU (M540) with 2.53 GHz and 4 GB RAM and a solid-state drive.

Tool/grid size (rows × columns)	ESRI ArcGIS ¹	TauDEM ²	SAGA GIS ³	Whitebox ⁴	TopoToolbox 2
1385 × 1371					
Fill sinks	3	3.1 (2.3)	4	5	0.22
D8 flow direction	6	24 (23)	–	2	2.51
Flow accumulation	44	1.5 (1.3)	5	3	0.04
2769 × 2742					
Fill sinks	12	10.5 (7.7)	17	19	0.90
D8 flow direction	16	157 (153)	–	5	10.1
Flow accumulation	188	4.2 (3.6)	18	8	0.15
5537 × 5484					
Fill sinks	43	30 (26)	66	80	4.03
D8 flow direction	50	1344 (1331)	–	18	40.3
Flow accumulation	363	15 (13)	58	29	0.66

¹ <http://www.esri.com/software/arcgis/extensions/spatialanalyst/>; version 10.1; benchmarking excludes default pyramid and statistics calculation and output data type is set to integer.

² <http://hydrology.usu.edu/taudem/taudem5/index.html>; version 5; we used the command line interface to run the application with 8 parallel processes. Numbers in parentheses refer to computation time without overhead time for reading and writing.

³ <http://www.saga-gis.org/>; version 2.1; fill sinks algorithm proposed by Wang and Liu (2006) (XXL) with minimum of slope of 0.001°; recursive catchment area algorithm.

⁴ <http://www.uoguelph.ca/~hydrogeo/Whitebox/>; version 3.0.8; fill sinks algorithm proposed by Wang and Liu (2006) with minimum of slope of 0.001°.

of the memory of the associated multiple-flow-direction matrix and 33 % of the single-flow-direction matrix. Because the nodes can be divided into independent subsets according to drainage basin affiliation, they can be distributed to different processors, so that flow-related terrain attributes can be evaluated in parallel, an important aspect for reducing computation times in LEMs (Braun and Willett, 2012).

In a topologically sorted FLOWobj, each grid cell's index ix appears before its downstream neighbor ixc if there is a directed link from ix to ixc . For routing algorithms that simulate flow through the directed network, the topological order entails that each link or edge is only traversed once, which simplifies flow computations and decreases running time to $O(n)$. For example, flow accumulation is calculated by sequentially updating an initial weight raster A with rows x cols = n cells connected by a number of nrEdges directed links. The links connect upstream cells addressed with the vector of linear indices ix with their respective downstream neighbors indexed by the vector ixc .

$$A = \text{ones}(\text{row}, \text{cols}); \quad (1)$$

for $i = 1 : \text{nrEdges}$,

$$A(\mathbf{ixc}(i)) = A(\mathbf{ix}(i)) + A(\mathbf{ixc}(i)). \quad (2)$$

end

Note that the calculation is performed by in-place operations (that is, the function input A is directly overwritten) which significantly reduces overhead memory of the algorithm. The flow direction object (FLOWobj) can be created

from an existing flow direction matrix (SFD or MFD) or a DEM. The latter is currently supported only for D8 flow routing, but allows users to simultaneously define if and how flow directions across flat terrain and topographic depressions are derived. Sinks and contiguous, flat areas in a DEM can be traversed using an auxiliary topography calculated by a gray-weighted distance transform algorithm (Soille et al., 2003; Metz et al., 2011; Schwanghart et al., 2013). The resulting flow directions tend to correspond more closely to actual flow paths compared to those that are derived from a DEM that was hydrologically conditioned using “flood filling” (Poggio and Soille, 2011).

4 Performance comparison

To evaluate the performance of TopoToolbox 2, we compared the runtimes of tools implemented in TopoToolbox with those of similar software. We emphasize that times to perform these computations fail to provide a concise, comparable measure since different programs pursue different strategies to store, process and access the data. TAUDem 5, for example, does not hold the entire DEM in the main memory but partitions it into domains that are processed in parallel (Tesfa et al., 2011) and thus shows its particular strength with data sets that would exceed main memory capacity. Unlike TopoToolbox 2 and most other programs that calculate flow accumulation based on a data set that contains flow directions, SAGA GIS calculates flow accumulation directly from the DEM.

We compared the performance between TopoToolbox 2 and ArcGIS 10.1, TauDEM 5, SAGA GIS 2.1 and Whitebox GAT 3.0.8. We used a $6^\circ \times 6^\circ$ tile of the 3 arcsec DEM made available by Ferranti (2013) covering the Alps and their foreland, which we reprojected into a Universal Transverse Mercator projection. We chose this region since it includes a high diversity of landscapes ranging from high-mountain areas to flat and low-lying topography and encompasses frequent sinks and flat areas that either arise from valley narrowing or lakes or as a result of the integer representation of elevations. The resulting DEM has 5537 rows and 5484 columns and a cell size of 79 m. We measured the times that each program requires (1) to hydrologically condition the DEM by filling sinks, (2) to derive D8 flow directions and (3) to calculate flow accumulation. We repeated this procedure after resampling the DEM to a cell size of 158 and 316 m, corresponding to 2769×2742 and 1385×1371 rows and columns, respectively.

Table 1 contains the benchmark results and shows that the elapsed times required by the software to perform the respective operations are highly variable. Sink processing is performed the fastest with TopoToolbox, which resorts to MATLAB's image-processing toolbox's built-in morphological reconstruction algorithm (Vincent, 1993). Besides the adoption of different algorithms by the other programs, reasons for not being as fast may include data swapping between the main memory and the hard disk to avoid memory problems (ArcGIS, TauDEM) or due to minimum slope imposition onto flat sections as implemented in SAGA GIS and Whitebox GAT. Whitebox GAT performs better in deriving D8 flow directions, which is likely attributed to TopoToolbox's computationally more demanding approach that enforces flow convergence in flat terrain (Soille et al. 2003). Finally, the new approach in TopoToolbox to represent flow directions leads to approximately 20 times faster calculation of flow accumulation than TauDEM 5, which we found to be second fastest among the programs that we compared.

5 Further changes and enhancements

The main goal of TopoToolbox is to provide access to numerical tools for DEM analysis in the form of functions that can be used as a library in custom scripts or functions for advanced analyses or modeling purposes. Most functions available with TopoToolbox 1 are now methods associated with the new classes GRIDobj and FLOWobj. Changes to calling syntaxes are few and were mainly made since the new classes encapsulate information such as cell size and spatial referencing in their properties, which previously had to be supplied to functions as additional input arguments. Thus, function syntaxes have become simpler and more intuitive. A new suite of functions was developed around the class STREAMobj that allows for creating, modifying and analyzing stream networks. Emphasis is placed on tools that facil-

itate the study of drainage networks and their relation to the tectonic setting, e.g., the analysis of longitudinal river profiles using slope–area plots (Wobus et al., 2006) or χ plots (Harkins et al., 2007; Perron and Royden, 2013; Scherler et al., 2013).

Additional performance increases were achieved by writing many flow-related algorithms as subroutines in C programming language executed by MATLAB (C-MEX), which provides 5–10 times faster evaluation of algorithms such as flow accumulation or drainage basin delineation. Computation speed is particularly significant when repeatedly calculating flow networks and accumulation areas, such as in numerical LEMs (e.g., Willgoose, 2005; Pelletier, 2008; Tucker and Hancock, 2010). While rivers act as major pathways for sediment or upstream-migrating knickpoints in mountain environments, they encompass only a small portion of the entire landscape. Accordingly, in LEMs the heterogeneous distribution of upslope areas leads to numerical solutions that require very small time steps to be numerically stable. Though implicit methods exist that relax the time step constraints (Fagherazzi et al., 2001; Braun and Willet, 2013), they fail to include the planform changes of river and flow network development, thus making frequent updating of the flow directions necessary. With its new implementation, TopoToolbox 2 can serve as a function library for the development of LEMs in MATLAB by providing fast-executing codes in a programming language that already supports a large library of numerical tools.

Visual exploration and interpretation of DEMs and data derived during their analysis constitute an integral work step in most studies. Furthermore, certain tasks are often easier and more rapidly accomplished through direct interaction with DEMs and their derivatives, as compared to writing and running scripts. Thus, TopoToolbox 2 now comes with several applications that allow usage and exploration of various toolbox functions with a GUI (Fig. 1).

The application topoapp is an explorative tool for visual analysis of DEMs and provides rapid access to many basic and several advanced functions of the TopoToolbox 2. Importantly, this application is defined as an own object class that, when working on an instance of topoapp with the GUI, remains in the workspace and is constantly updated with every action of the user. Additionally, it was designed in a way that allows users to add their own functions or modules to it and therefore provides an expandable platform for displaying the graphical outcomes of DEM analysis with the TopoToolbox and other MATLAB tools. With the currently embedded functions, users can quickly and easily extract and plot simple profiles; entire drainage networks, or subsets of them; watershed boundaries; and individual channel reaches (Fig. 1). The created features can be used to build swath profiles, to subset a DEM, to compute basic catchment statistics and to create χ plots; furthermore, all features can also be exported to common formats, amongst other options.

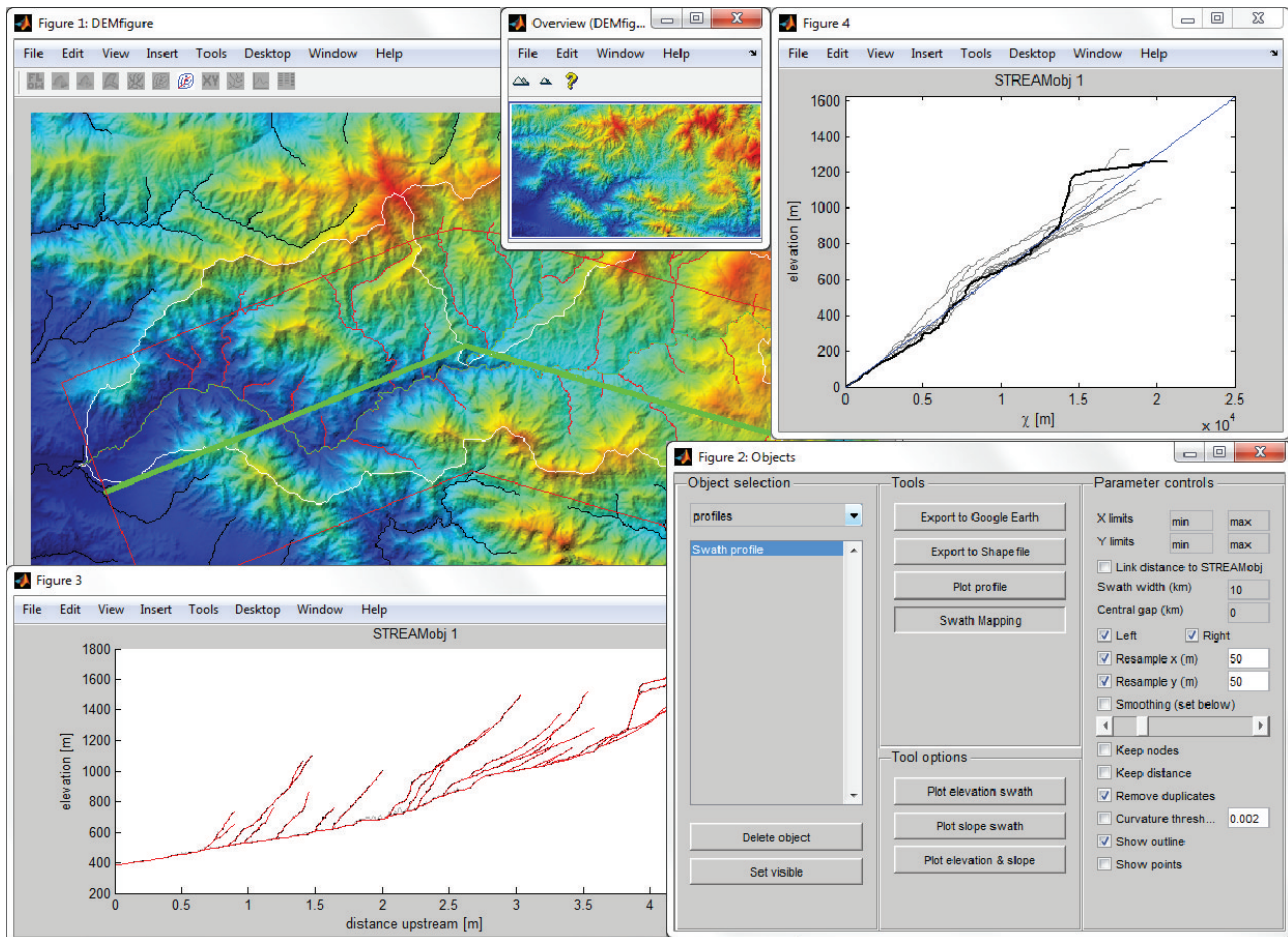


Figure 1. Example layout of topoapp, a graphical user interface that enables access to the majority of TopoToolbox functions.

Manual editing of DEMs with spurious sinks is often required to produce hydrologically correct DEMs. Although automated ways to distinguish between actual and spurious sinks exist (Lindsay and Creed, 2006), choosing which sinks should be filled, carved or retained may be preferred and requires visual interpretation of the data and results. The new application preprocessapp enables a combination of automatic and manual hydrological correction of a DEM by providing interactive tools to fill topographic depressions or carve along least-cost paths to breach features that obstruct digital flow paths. The application flowpathapp is a tool to map individual streams or a stream network by interactively setting channel head locations, and slopeareatool provides an interactive tool to study slope–area relationship for individual streams or stream networks.

6 Conclusions

With the release of version 2, the TopoToolbox for MATLAB has become a more memory-efficient and faster tool for digital terrain analysis. The main difference to the previous ver-

sion is the adoption of an object-oriented representation of grids, flow direction and stream networks. Topological ordering reduces the memory bottleneck of the previous, sparse matrix approach to flow direction while increasing computation speed and allowing parallel computing. Software quality and particularly reusability are improved so that TopoToolbox can be used as a framework for implementing dynamic, hydrological and geomorphological models. Finally, newly introduced, interactive tools will not only be valuable for researchers to perform rapid analysis but also for conveying and exploring the richness of Earth surface processes in classrooms or labs.

TopoToolbox 2 is platform-independent, free and open software, but requires MATLAB 2011b or later as well as the Image Processing Toolbox. Version 2.0 is included in the supplementary material. The software, future releases and updates are hosted by the Community Surface Dynamics Modeling System (CSDMS) server (<http://csdms.colorado.edu/>).

Supplementary material related to this article is available online at <http://www.earth-surf-dynam.net/2/1/2014/esurf-2-1-2014-supplement.zip>.

Acknowledgements. W. Schwanghart is supported by the Potsdam Research Cluster for Georisk Analysis, Environmental Change and Sustainability (PROGRESS). D. Scherler is supported by a scholarship from the Alexander von Humboldt Foundation and by Caltech's Tectonics Observatory. The support is gratefully acknowledged. We thank P. Steer, T. Hengl and A. J. A. M. Temme for their constructive comments on a previous version of the manuscript.

Edited by: A. Temme

References

- Arge, L., Chase, J. S., Halpin, P., Toma, L., Vitter, J. S., Urban, D., and Wickremesinghe, R.: Efficient flow computation on massive grid terrain datasets, *GeoInformatica*, Springer Netherlands, 7, 283–313, 2003.
- Borghi, A., Renard, P., and Jenni, S.: A pseudo-genetic stochastic model to generate karstic networks, *J. Hydrol.*, 414/415, 516–529, 2012.
- Braun, J. and Willett, S. D.: A very efficient $O(n)$, implicit and parallel method to solve the stream power equation governing fluvial incision and landscape evolution. *Geomorphology*, 180/181, 170–179, 2013.
- Chen, M. and Jacquemin, M.: Footprints of dependency: towards dynamic memory management for massive parallel architectures. Technical report, Yale University, Department of Computer Science, YALEU/DCS/TR-593, 1988.
- Clason, C., D. Mair, W. F., Burgess, D. O., and Nienow, P. W.: Modelling the delivery of supraglacial meltwater to the ice/bed interface: application to southwest Devon Ice Cap, Nunavut, Canada. *J. Glaciol.*, 58, 361–374, 2012.
- Costa-Cabral, M. and Burges, S.: Digital Elevation Model Networks (DEMON), a model of flow over hillslopes for computation of contributing and dispersal areas. *Water Resour. Res.*, 30, 1681–1692, 1994.
- De Moel, H., Asselman, N. E. M., and Aerts, J. C. J. H.: Uncertainty and sensitivity analysis of coastal flood damage estimates in the west of the Netherlands. *Nat. Hazard Earth Sys.*, 12, 1045–1058, 2012.
- Dietrich, W. E., Bellugi, D. G., Sklar, L. S., and Stock, J. D.: Geomorphic transport laws for predicting landscape form and dynamics. *Prediction in Geomorphology*, American Geophysical Union, 135, 1–30, 2003.
- Fagherazzi, S., Howard, A. D., and Wiberg, P. L.: An implicit finite difference method for drainage basin evolution. *Water Resour. Res.*, 38, 21.1–21.5, 2002.
- de Ferranti, J.: Digital elevation data, Url: <http://www.viewfinderpanoramas.org/dem3.html>, accessed on 20 October 2013.
- Gruber, S. and Peckham, S.: Land-surface parameters and objects in hydrology. In: Hengl, T., Reuter, H. I. (Eds.), *Geomorphometry. Concepts, Software, Applications*, Elsevier, 33, 171–194, 2009.
- Harkins, N., Kirby, E., Heimsath, A., Robinson, R., and Reiser, U.: Transient fluvial incision in the headwater of the Yellow River, northeastern Tibet, China. *J. Geophys. Res.*, 112, F03S04, doi:10.1029/2006JF000570, 2007.
- Huang, P.-C. and Lee, K. T.: An efficient method for DEM-based overland flow routing. *J. Hydrol.*, 489, 238–245, 2013.
- Jasiewicz, J.: A new GRASS GIS fuzzy inference system for massive data analysis. *Comput. Geosci.*, 37, 1525–1531, 2011.
- Kepner, J.: Graphs and matrices, in: *Graph algorithms in the language of linear algebra*, SIAM series on Software, edited by: Kepner, J. and Gilbert, J., Environments and Tools, Philadelphia, 3–12, 2011.
- Lindsay, J. B. and Creed, I. F.: Distinguishing actual and artefact depressions in digital elevation data. *Comput. Geosci.*, 32, 1192–1204, 2006.
- Messier, K.P., Akita, Y., and Serre, M. L.: Integrating address geocoding, land use regression, and spatiotemporal geostatistical estimation for groundwater tetrachloroethylene. *Envir. Sci. Tech. Lib.*, 46, 2772–2780, 2012.
- Metz, M., Mitasova, H., and Harmon, R. S.: Efficient extraction of drainage networks from massive, radar-based elevation models with least cost path search. *Hydrol. Earth Syst. Sci.*, 15, 667–678, 2011, <http://www.hydrol-earth-syst-sci.net/15/667/2011/>.
- O'Callaghan, J. F. and Mark, D. M.: The extraction of drainage networks from digital elevation data. *Comput. Vision Graph.*, 28, 323–344, 1984.
- Pelletier, J.: *Quantitative modeling of Earth surface processes*, Cambridge University Press, 295 pp., 2008.
- Perron, J. and Royden, L.: An integral approach to bedrock river profile analysis. *Earth Surf. Proc. Land.*, 38, 570–576, 2013.
- Pike, R., Evans, I., and Hengl, T.: *Geomorphometry: A brief guide*. In: Hengl, T., Reuter, H. I. (Eds.), *Geomorphometry. Concepts, Software, Applications*, Elsevier, 33, 3–30, 2009.
- Poggio, L. and Soille, P.: Influence of pit removal methods on river network position. *Hydrol. Process.*, 26, 1984–1990, 2011.
- Quinn, P., Beven, K., Chevallier, P., and Planchon, O.: The prediction of hillslope flow paths for distributed hydrological modelling using digital terrain models. *Hydrol. Process.*, 5, 59–79, 1991.
- Register, A. H.: *A guide to MATLAB object-oriented programming*. Chapman & Hall/CRC, SciTech Publishing, Boca Raton, 354 pp., 2007.
- Scherler, D., Bookhagen, B., and Strecker, M. R.: Tectonic control on ^{10}Be -derived erosion rates in the Garhwal Himalaya, India. *J. Geophys. Res. Earth Surf.*, in press, 2013.
- Schwanghart, W. and Kuhn, N. J.: TopoToolbox: A set of Matlab functions for topographic analysis, *Environ. Modell. Softw.*, 25, 770–781, 2010.
- Schwanghart, W. and Jarmer, T.: Linking spatial patterns of soil organic carbon to topography – a case study from south-eastern Spain, *Geomorphology*, 126, 252–263, 2011.
- Schwanghart, W., Groom, G., Kuhn, N. J., and Heckrath, G.: Flow network derivation from a high resolution DEM in a low relief, agrarian landscape. *Earth Surf. Proc. Land.*, 38, 1576–1586, 2013.
- Shahzad, F. and Gloaguen, R.: TecDEM: A MATLAB based toolbox for tectonic geomorphology, Part 1: Drainage network pre-processing and stream profile analysis. *Comput. Geosci.*, 37, 250–260, 2011.

- Soille, P.: Morphological carving. *Pattern Recogn. Lett.*, 25, 543–550, 2004.
- Soille, P., Vogt, J., and Colombo, R.: Carving and adaptive drainage enforcement of grid digital elevation models. *Water Resour. Res.*, 39, SWC 10.1–13, 2003.
- Tarboton, D. G.: A new method for the determination of flow directions and upslope areas in grid digital elevation models. *Water Resour. Res.*, 33, 309–319, 1997.
- Tesfa, T. K., Tarboton, D. G., Watson, D. W., Schreuders, K. A., Baker, M. E., and Wallace, R. M.: Extraction of hydrological proximity measures from DEMs using parallel processing. *Environ. Modell. Softw.*, 26, 1696–1709, 2011.
- Tucker, G. E. and Hancock, G. R.: Modelling landscape evolution. *Earth Surf. Proc. Land.*, 35, 28–50, 2010.
- Vincent, L.: Morphological grayscale reconstruction in image analysis: applications and efficient algorithms, *IEEE T. Image Process.*, 2, 176–201, 1993.
- Wang, L. and Liu, H.: An efficient method for identifying and filling surface depressions in digital elevation models for hydrologic analysis and modelling. *Int. J. Geogr. Inf. Sci.*, 20, 193–213, 2006.
- Willgoose, G.: Mathematical modelling of whole landscape evolution. *Annu. Rev. Earth Pl. Sc.*, 33, 443–459, 2005.
- Wilson, J. P. and Gallant, J. C.: Digital terrain analysis, in: *Terrain Analysis: Principles and Applications*, edited by: Wilson, J. P. and Gallant, J. C., Wiley, 1–27, 2000.
- Wobus, C., Whipple, K. X., Kirby, E., Snyder, N., Johnson, J., Spyropoulou, K., Crosby, B., and Sheehan, D.: Tectonics from topography: procedures, promise, and pitfalls, *GSA Special Papers*, 398, 55–74, 2006.