

## New computer program for digital terrain analysis

Vojtěch Barták

Department of Applied Geoinformatics and Spatial Planning, Faculty of Environmental Sciences  
University of Life Sciences Prague, Kamýcká 129,  
165 21, Prague, Czech Republic  
bartakv@fzp.czu.cz

**Abstract.** Raster digital elevation models (DEMs) represent a source of data with rapidly increasing quality and availability, and are usable in many fields of research and planning, especially in hydrological modelling and river basin management. Their increasing use should go hand in hand with the availability of adequate techniques for their processing. Unfortunately, there is still some gap between the development of digital terrain analysis (DTA) algorithms and their accessibility in commonly used (commercial or non-commercial) GIS software. Such programs (e.g. ArcGIS, GRASS etc.) contain often merely one or few algorithms for a particular task, moreover sometimes the oldest and worst ones (e.g. the Jenson and Domingue's (1988) algorithm for removing spurious pits from DEM that is implemented in ArcGIS). In this paper, a new DTA computer program DEMETERR 1.1 (an acronym for Digital Elevation Models and their Elementary TERRain analysis) is presented. Apart from the computation of the general terrain attributes (terrain slope, aspect, horizontal and vertical curvature etc.) it contains a number of algorithms for most of the basic DTA tasks in hydrological applications, such as removing closed depressions (often called pits or sinks), treatment of flat areas, drainage direction assignment, contributing areas and topographic index computation, watershed delineation and river network extraction. The available methods cover most of the algorithms described in literature, and the program enables one to try different methods of solving the same problem, to compare the results and to choose the best method for his purpose. The program has a user-friendly graphical interface with the possibility of viewing and saving the textual and graphic outputs. Further development of the program will contain the improvement of the input/output format options.

**Keywords:** Digital terrain analysis, Raster digital elevation models, DEMETERR, Hydrological modelling, GIS, Overland flow algorithms

### 1 Introduction

The increasing use of raster digital elevation models (DEMs) as a source of complex information about catchments should go hand in hand with the development and availability of adequate DEM processing techniques. Unfortunately, there is still a wide gap between the development of digital terrain analysis (DTA) algorithms and their accessibility in commonly used (commercial or non-commercial) GIS software. Such programs (e.g. ArcGIS, GRASS etc.) contain often merely one or few algorithms for a particular task, moreover sometimes the oldest and worst ones (e.g. the Jenson and Domingue's [9] algorithm for removing spurious pits from DEM that is implemented in ArcGIS).

The major aim of the presented program is to contribute to filling this gap by providing a useful tool for resolving all the elementary (hydrologically oriented) DTA tasks. Implemented methods for the particular tasks mostly cover the majority of algorithms described in literature, so the modeller can use them and consequently compare the results. Most of the presented algorithms have been developed by hydrologists for the purpose of modelling the outflow from catchments and the river basin management.

The structure of the paper is as follows: Chapter 2 provides a brief outline of the major DTA tasks and the methods for their resolving. Apart from the basic terrain attributes, the major tasks involve flow direction assignment, catchment delineation, and river network extraction. Since the solution of such tasks is usually based on the preceding flow directions identification, it is usually desirable to treat the problematic parts of DEM, such as closed depressions and flat areas, where the water cannot flow downhill and the flow direction remains undefined.

Chapter 3 describes the program itself, its inputs and outputs, the graphical user interface, and, especially, the choice of tasks and algorithms. In the forth chapter, the accessibility and orientation of a further development and improvement of the program is outlined.

Due to the amount of tasks and methods discussed here, I shall describe each of them very briefly, often without investigating the important details, advantages and disadvantages. The interested reader may turn to the bibliography.

## 2 Major DTA tasks and methods

### 2.1 Primary and secondary terrain attributes

In this outline, I shall use the definitions and the classification from the paper by Moore et al. [21]. These authors distinguish between primary and secondary terrain attributes derivable from a DEM. The primary ones are computed directly from elevations, typically, using the values of its eight neighbours (i.e. the surrounding cells) for each DEM cell. The examples may be the terrain slope, the aspect, the (horizontal or vertical) curvature, or the contributing area (often called the upslope drainage area or the flow accumulation) defined as the upslope area draining through the particular cell (i.e. the catchment of the cell).

The slope, aspect and curvature can be calculated for the particular cell from the elevations of all its eight neighbours, using the first and second discrete derivatives (for details see for example [4]). Another way is to compute the slopes between the cell and its neighbours (as the ratio between the vertical and horizontal distance) and then use the steepest one (i.e. greatest downhill slope), or their weighted mean where the weights are proportional to the slope magnitude [24]. These methods directly correspond to certain methods of contributing areas computing, which are described in the next section.

Secondary terrain attributes are defined as a compound of the primary attributes. The well-known example is the topographic index (sometimes called the wetness index) indicating the tendency of the site to soil saturation [25]. It is defined as a  $\log(a/S)$ , where  $a$  is the contributing area of the cell and  $S$  is the slope. For the comprehensive list of primary and secondary terrain attributes see e.g. [21].

### 2.2 Overland flow simulation

In the present context, the term “overland flow” does not mean any real outflow process, but the theoretical water flow over a completely uncovered and impermeable terrain, generated by the initial unit rainfall regularly spaced over the entire model. The first step is to determine the direction for each cell, along which the flow will proceed to one or several of its neighbours. When the flow directions are assigned, the flow accumulation at each cell can be computed as the number of cells which drain through the flow directions to this cell. Such values multiplied by the cell area are often called “contributing areas” or “upslope drainage area” (the last term reflects the possibility of computing the “downslope drainage areas” whose definition is analogous).

Before discussing the methods of flow accumulation computing, several techniques of flow directions assignment will be presented. Some other methods can be found in literature (see an extensive review by [1]); however, here I have chosen the most suitable ones.

**SFD8.** The first references to the SFD8 (Single Flow Direction chosen from 8 possibilities) algorithm are probably in [14] and [23]. The algorithm simply determines the flow direction from a given cell to that of its neighbours, to which the slope is steepest (see Figure 1). The slope from the cell to its neighbour is computed as the elevation difference divided by the horizontal distance (cell size for cardinal neighbours and cell size multiplied by square root of 2 for diagonal neighbours), and is considered as positive downhill.

7	9	8
6	5 → 2	3
3	4	1
1.41	1	2.83

**Figure 1.** The SFD8 algorithm. The black numbers denote the elevations, the small green numbers denote the slope from the central cell, and the red arrow represents the determined flow direction.

The basic feature of this approach is that flow is always represented as convergent. No splitting of the flow to several directions is allowed. This may be regarded as an advantage as well as a

disadvantage, depending on the aim of the analysis. The additional and clearly undesirable feature is that the flow is biased towards one of the eight possible directions, so the result is strongly dependent on the grid orientation of the DEM.

**MFD8.** The MFD8 (Multiple Flow Direction chosen from 8 possibilities) algorithm was developed as a modification of the SFD8 to represent flow divergence. Water flow from the cell is divided to all the neighbours to which exists a positive (i.e. downhill) slope (see Figure 2). The proportion of flow assigned to  $i$ 'th neighbour is computed using the ratio

$$\frac{S_i^p}{\sum_{0 < j \leq 8} S_j^p}, \quad (1)$$

where  $S_i$  is the slope from the cell to its  $i$ 'th neighbour, the summation is over the neighbours to which a positive slope exists, and  $p$  is a parameter, which can be considered as constant or spatially variable (for the latter see [25] and [11]). The original formulation of (1) with  $p = 1.1$  was proposed by Freeman [3], and with  $p = 1$  and slopes multiplied by so called "contour length" (coefficient greater for diagonal than for cardinal neighbours) by Quinn et al. [24]. The generalization to  $p \neq 1$  (or 1.1) was given by Holmgren [7] and Quinn et al. [25]. Note that the higher the values of the parameter  $p$ , the greater the similarity of the MFD8 algorithm to the SFD8 (theoretically if  $p \rightarrow \infty$  then both algorithms merge). In other words, the exponent  $p$  controls the degree of flow convergence.

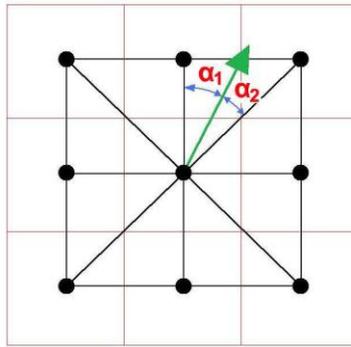


**Figure 2.** The MFD8 algorithm. The thickness of the red arrows symbolizes the proportions of flow falling on the particular neighbour.

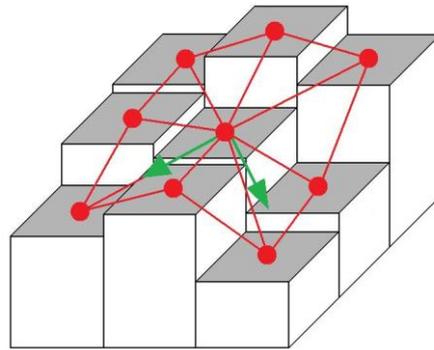
**SFD $\infty$ .** This approach (Single Flow Direction chosen from infinite possibilities) was proposed by Tarboton [32]. The algorithm chooses the direction of the steepest slope from the (theoretically) infinite range from  $0^\circ$  to  $360^\circ$ . It proceeds by constructing eight triangles with vertices in the centres of the cell and its two (mutually adjacent) neighbours (see Figure 3), and subsequently identifies the steepest slope direction on the planes determined by the triangles. The greatest of such slopes is then chosen and, if it doesn't go directly to the centre of one of the neighbours, the flow is apportioned to the corresponding two neighbours, so that the amount of flow falling on the particular neighbour is inversely proportional to the deviation of the direction to the neighbour from the determined steepest slope direction (the angles  $\alpha_1$  and  $\alpha_2$  on the Figure 3).

The fact that the flow is mostly divided into two directions led some authors to consider this algorithm as "multiple flow" with the representation of divergent flow. Nevertheless, the original aim [32] was to represent flow convergence, but without the undesirable bias towards cardinal and diagonal directions. The flow divergence obviously occurs, but it is always constrained to the two adjacent neighbours.

**MFD $\infty$ .** The method (Multiple Flow Direction chosen from infinite possibilities), developed by Seibert et McGlynn [28], is a straightforward generalization of the SFD $\infty$ , in the sense that the MFD8 is a generalization of the SFD8. As in the SFD $\infty$ , the algorithm seeks the steepest slope directions on the eight triangle-generated planes. Then, however, it doesn't choose the steepest one but all the positive and locally steepest ones (see Figure 4). The flow apportioning between the chosen directions is realized by (1), and the corresponding flow part for each such direction is subsequently divided into two mutually adjacent neighbours in the same manner as in the SFD $\infty$ .



**Figure 3.** The  $SFD^\infty$  algorithm. The black dots are the centres of the cells, the green arrow is the determined flow direction. Note that this example does not represent the topography from Figures 1 and 2 but some other.



**Figure 4.** The  $MFD^\infty$  algorithm. The red lines and dots delineate the triangles, on which the locally steepest slopes are determined. The green arrows represent the determined flow directions.

Once the flow direction is defined for each cell, one can simulate the flow accumulation and compute the contributing areas. One possible method [23] is as follows. At the beginning, the flow accumulation value is set to 1 for each cell. Then the DEM is scanned by row-by-row scan and, for each cell, the number of its inflows (i.e. the neighbours whose flow direction is directed towards it) is counted. Subsequently, the DEM is repeatedly scanned (row-by-row) and, during each iteration, the cells with no inflow are determined and their flow accumulation values are distributed among their neighbours (by one of the methods described above). Simultaneously, the numbers of inflows of such neighbours are decreased by one. This implementation has the  $O(n^2)$  asymptotic complexity. The complexity  $O(n)$  can be obtained when the neighbours whose number of inflows dropped to zero are stored in the stack and the computation then proceeds at the first cell on the stack (while the stack is nonempty; see [1]). Thus the iterative row-by-row scan is avoided. Another approach, with asymptotic complexity  $O(n \log n)$ , uses the priority queue and searches the DEM “from highest to lowest”.

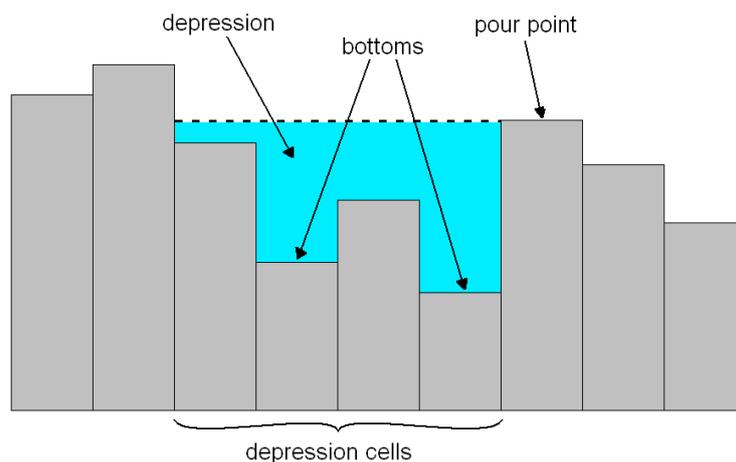
### 2.3 Catchment delineation

For delineating a catchment, the position of the outlet must be known. Such outlet may be represented by a single cell or by a group of cells. The catchment consists of all the cells draining through the flow directions to the outlet. Finding such cells can be realized exactly by the same way as the flow accumulation described in the previous section, with two differences. The first is that only the directions are traced and no flow partitioning must be carried out. The second is that the output is the list of the cells belonging to the catchment rather than their count.

## 2.4 Closed depressions removal

Closed depressions (often called sinks or pits) are cells or groups of cells completely surrounded by cells with higher elevation. It means that no flow direction can be assigned to such cells, since no positive slope is determined. Most landscapes are modelled by fluvial and erosion processes, hence the presence of closed depressions in the DEM is usually due to data errors, such as the errors from interpolation or measurement [2]. Although sometimes they can reflect the real terrain features (e.g. the volcano craters), modellers often deal with them as with spurious features that must be removed from the DEM, otherwise they cause discontinuities in the drainage network determined by flow directions, and thus prevent the successful solution of other DTA tasks (e.g. the catchment delineation or the river network extraction). An interesting attempt to develop a method for distinguishing between the actual and the spurious depressions is presented in [13].

Before describing methods for the depressions removal, it is necessary to discuss their identification in the DEM, since it usually must be done before their removal, and since various complex depressions, such as nested or cascade-organized depressions, make this procedure surprisingly complicated. The discussion cannot be made without some terminology. Each depression has a “bottom” (consisting of cells with no lower neighbour), and a “catchment” (consisting of cells from which the flow paths terminate at the bottom). The lowest cell on the boundary of a catchment (i.e. the point through which the water will overflow when the depression is filling) is often called “pour point”, and the depression itself can be defined as consisting of all the cells from its catchment whose elevation is lower than the elevation of the pour point (see Figure 5).



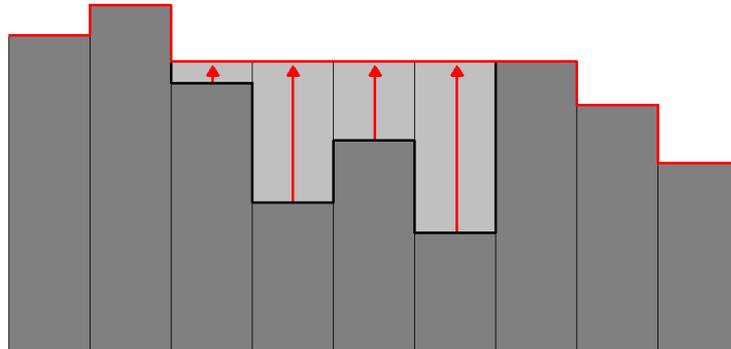
**Figure 5.** The vertical section of a DEM with depression.

The oldest methods for finding the depressions are based on the following two-phase procedure ([9], [15]). In the first phase, the bottoms are found by row-by-row scan; in the second phase, for each bottom the catchment (and the pour point on its boundary) is found. This type of procedure has the asymptotic complexity  $O(n^2)$ , where  $n$  is the number of cells in the DEM, and since they treat each depression separately, their real complexity depends on the number and complexity of depressions.

Soille et Gratin [30], and later (without any reference to their paper!) also Wang et Liu [40], suggested a more effective procedure. It uses the “from lower to higher” and “from edges inwards” search, based on the priority queue data structure (also known as binary heap), rather than the row-by-row scan. Only one search is needed and all the depressions and pour points are easily identified, with asymptotic complexity  $O(n \log n)$  (the complexity of priority queue).

**Filling.** The simplest and probably best known method for removing the depressions from the DEM, firstly proposed by Jenson et Domingue [9] and Martz et DeJong [15], is to “fill” them by increasing the elevations of the depression cells up to the elevation of pour point (see Figure 6). Its theoretical background is the assumption that the depressions are caused by an underestimation of the elevations in the depressions. Filling the depressions can be effortlessly performed during the priority-queue-based depression search described above ([30], [40]).

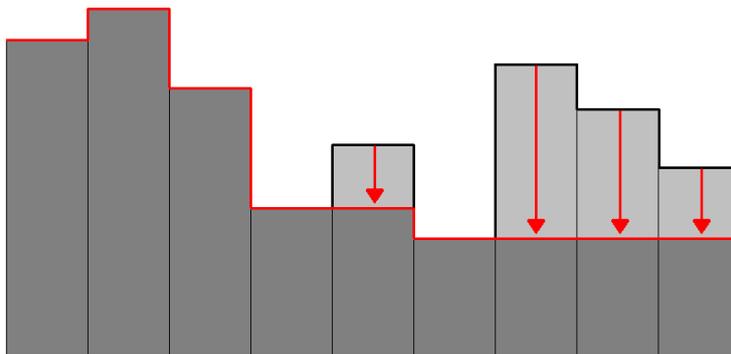
The obvious disadvantage of this method is that the filled depressions create large flat areas, where the problem with flow direction assignment arises again. Such regions must be further treated by some method for flats removing (see the next section).



**Figure 6.** An example of depression filling. The black thick line is the original topography before filling, the red line is the new topography after filling.

**Carving.** The alternative assumption about the origin of depressions can be that depressions are due to the overestimation of elevations in the neighbourhood of the pour point, where the overestimated cells formed a “dam”. The proper solution is therefore to decrease rather than increase the elevations of the affected cells (Figure 7). This approach (sometimes called “breaching”) was proposed independently (and with some differences) by Rieger [26], Jones [10] and Soille et al. [31]. The latest implementation is also based on a priority-queue-search and thus shares its complexity.

Although sometimes this method is adequate and removes the depressions by decreasing the elevations of only few cells, in the case of small and deep depressions it can sometimes cause the unrealistic, long and deep “channels” consisting of the decreased cells. In such cases, the filling approach would probably be better.

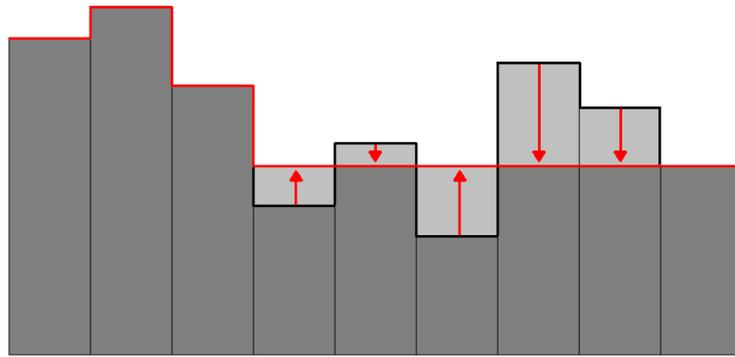


**Figure 7.** An example of depression carving. The black thick line is the original topography before carving, the red line is the new topography after carving.

**Combinations.** The logical extension of the described approaches is the combination of both, with the particular method being chosen according to its impact (or cost). The cost may be measured as the amount of the changed cells (the area cost), as the total change in elevations (the volume cost), or as a combination of these two. The last possibility is adopted in the IRA (Impact Reduction Approach) method proposed by Lindsay et Creed [12], in which the cost is computed for each of the two methods (filling as well as carving) applied to the whole DEM, and then the method with the lower cost is chosen.

A straightforward improvement of such method would be to decide between the filling and the carving for each depression separately. Unfortunately, this cannot be done easily due to the presence

of complex depressions [1]. However, there are two methods that solve each depression separately, and, moreover, enable us to treat some part of the depression by carving and the rest by filling (see Figure 8). The first method was proposed by Martz et Garbrecht [17], who called it “outlet breaching” (sometimes it is called “constrained breaching”). The procedure tries to decrease the elevations of some cells in the neighbourhood of the pour point, before the filling is performed. Soille [29] suggested a more sophisticated method based on the mathematical morphology, in which the proportion of carving and filling is set in such manner that the resulting cost is the lowest possible.



**Figure 8.** Combination of filling (arrows up) and carving (arrows down).

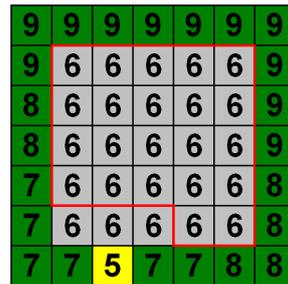
**Physically based methods.** These methods try to create a terrain which is more realistic than the flat areas generated by filling or carving procedures. The first and very simple approach is to fill the depressions in a manner similar to the filling technique, but the elevations of the cells in the depression are not increased at the pour point elevation but slightly higher. The method is called SDI (Spatially Distributed elevation Increment; [36]), which reflects the fact that the increment added to the pour point elevation is spatially distributed in order to generate a gently inclined plane. The important assumption is that the DEM is relatively simple and roughly inclined to one of its corners, so the resulting slope orientation of the plane originated from the particular depression corresponds to this DEM inclination.

The second and truly physically based method is PEM4PIT (Physical Erosion Model for PIT filling; [6]). It simulates the filling-up of the depressions by erosion, the intensity of which is consistent with the surrounding topography. The elevations of the cells in the depressions are changed so that the new topography satisfies the mass continuity equation for steady state topography, reflecting the equilibrium between tectonic, fluvial and diffusion processes in landscape.

Despite the generation of more realistic terrains, both methods have the awkward feature that they are usually iterative, since during the raising elevations some new (mostly small and shallow) depressions may occur. This leads to a relatively high cost, containing elevation changes on the cells which originally were not in any depression. On the other hand, the advantage of such methods is that the flat areas and the depressions can be treated simultaneously by the same procedure.

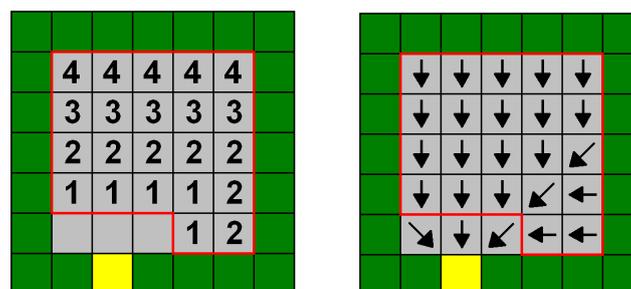
## 2.5 Flat areas treatment

Flat areas (or “flats”) are cells or groups of cells completely surrounded by cells with the same (or higher) elevation (see Figure 9). They are often present in DEMs due to data rounding on relatively flat topography, or as the result of a particular depression-removing procedure. Apart from the methods which treat flats together with depressions (both physically based methods described above and the carving implementation proposed by Jones [10]), literature offers two other usable approaches, both constructing the improved topography of the flat according to the surrounding topography.

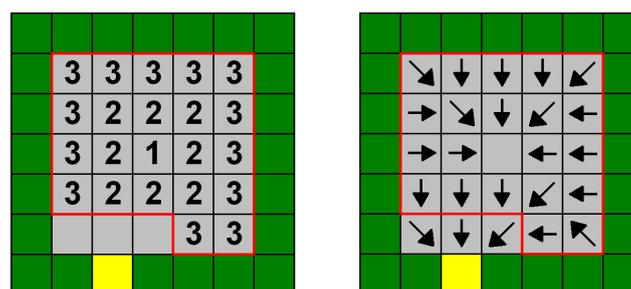


**Figure 9.** An example of the flat. The numbers denote elevations, the flat is bordered by the red line. The grey cells have the flat elevation, the green cells are higher and the yellow cell is lower than the flat elevation.

**Towards lower gradient.** The first method, originally proposed by Jenson et Domingue [9] and improved by Soille et Gratin [30], is to construct a topography inclined towards the lower neighbours of the flat (see Figure 10). In the first step, the cells not adjacent to any lower cell are increased by a sufficiently small increment. This step is then repeated, while any cell with no lower neighbour exists. The increment must be determined so the elevations in the flat will not exceed the elevation range of the surrounding topography.



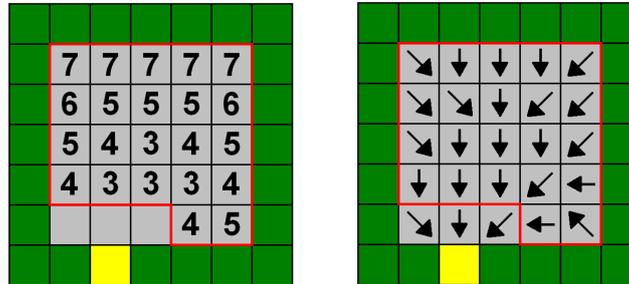
**Figure 10.** Towards lower gradient computed on the flat from Figure 9. On the left picture the numbers denote the number of elevation increments added to the particular cell. On the right picture the resulting flow directions (determined by the SFD8 algorithm) are shown.



**Figure 11.** From upper gradient computed on the flat from Figure 9. On the left picture the numbers denote the number of elevation increments added to the particular cell. On the right picture the resulting flow directions (determined by the SFD8 algorithm) are shown.

**Combined gradient.** The obvious result of the previous method is the simple inclined plane, in which the subsequently determined flow paths form non-realistic parallel lines (see Figure 10). To represent some degree of flow convergence, Garbrecht et Martz [5] suggested a method combining the gradient “towards lower” topography with the gradient “from higher” topography. The first is computed in exactly the same way as in the preceding method. The computation of the second is analogous: the algorithm iteratively increases the cells which are adjacent to some higher cell (and, in the first iteration only, are not adjacent to any lower cell; see Figure 11). The final step is the summation of both gradients (i.e. of

the number of increments added to each cell through any of the preceding steps, see Figure 12). Soille et al. [31] proposed an improved version of this algorithm, based on mathematical morphology, which implicitly solves certain special cases requiring an additional step in the original version (see [5] and [31]).



**Figure 12.** The final step of the combined gradient algorithm. The numbers in the left picture are sums of the numbers from Figures 10 and 11. On the right picture the resulting flow directions are shown. The flow direction of the central cell was determined through an additional step, since after summation of the total increments a new flat arised on this cell (see [5]). Such additional step simply consists of computation of the towards lower gradient on this new flat, with the half elevation increment (i.e. the number of increments added to the central cell would be 3.5). Soille et al. [31] proposed an improved version of this algorithm that does not require such additional step.

## 2.6 River network extraction

Sometimes the estimation of river network position from the DEM may be desirable, even when the “real” position (i.e. the topographic map) is known, because the real position often doesn’t agree with the DEM topography. Spatially distributed hydrological models that compute the outflow from a catchment using flow directions determined on the DEM cells should distinguish between “river cells” and “hill cells”, and represent the outflow from these two cell types in a different way, e.g. with flow divergence (multiple flow algorithms) on the hills and flow convergence (single flow algorithms) in the streams. Even in the river basin management, when subcatchment delineation is required, information about the position of the outlets (and thus the position of the river network) is necessary. In all these tasks, the position of the river cells must be compatible with the remaining DEM topograph so that the water will not be forced to flow uphill.

Despite of its real nature, the river network estimated from the DEM typically consists of lines of the width of one cell. Thus the methods for river network extraction must decide for each cell whether it is a river cell or not. Such decision may be done using the real river network position (the so called “blue lines” in the maps), or may be fully automatic, using the DEM as an exclusive source of information. The first approach, often called “enforcement” or “burning” (see e.g. [31], [38]), proceeds by overlaying the blue lines over the DEM and changing the elevations of the cells intersecting with blue lines. In this paper, the latter, fully automatic, approach is discussed.

**Local curvature methods.** The oldest methods for fully automated river network extraction are based on satisfying some criteria of the terrain shape typical for river valleys. Each cell is analyzed within the moving window (usually of the size of 3x3 cells), and the shape of the corresponding terrain is examined. One possible way is to make several vertical cuts in different directions and then to analyze these cuts as to whether they are in the form of the letter “V”, which is assumed to determine a river valley. Another way is to measure the degree of terrain convergence on each cell by counting the number of its neighbours that are higher then the inspected cell. The best known example is the Peucker&Douglas algorithm that scans the DEM by 2x2 sized window, and within each group of four cells the highest cell is marked. Then the river cells are determined as the cells that remain unmarked.

An extensive review of local curvature methods along with a detailed discussion is provided by Tribe [37], who noticed that these apparently simple methods often require an additional step, which makes them much more complicated. The typical resulting networks are namely discontinuous, full of inappropriately determined river cells and too wide so the necessary step consists in connecting (e.g. using formerly determined flow directions) and thinning them. An interesting algorithm solving all these obstacles was proposed by [18].

**Flow accumulation methods.** The basic concept of this approach is that river cells can be determined as cells in which the flow accumulation (i.e. the contributing area) is high enough for the transition of hillslope processes to fluvial processes [14]. Therefore the cell may be considered as a river cell if its contributing area exceeds some predefined threshold value. If the SFD8 algorithm is used, the obvious advantage of such approach will be that the resulting river network is fully connected. The particular methods differ in the way the threshold value is determined.

The simplest method is to consider the area threshold as spatially constant ([14], [9], [16], [25], and many others). The corresponding theoretical background was given by Tarboton et al. [34]. Such approach was criticized for an insufficient density of the resulting networks in the higher parts of a landscape and for the inappropriate position of river initiations (see e.g. [20]). One possible solution is to consider the area threshold as dependent on the slope values. Generally, the higher the slope, the lower the sufficient area threshold. In practice, several authors ([20], [19], and others) argue that the appropriate variable whose critical value has to be determined is  $aS^2$ , where  $a$  is contributing area and  $S$  is slope. This threshold often leads to discontinuous networks so the subsequent network connecting according to flow directions is usually required.

The substantial disadvantage of the described approach is that the proper estimation of a local slope is a very difficult task due to its strong scale dependency. Therefore Roth et al. [27] suggested to replace the slope by the so called "relative subcatchment elevation" that is defined for each cell as the mean elevation over its contributing area (i.e. its catchment) minus the elevation of the cell. They argue that this variable exhibits a behaviour similar to that of the local slope multiplied by the contributing area; however, its estimation is less affected by uncertainty due to its summation over a larger area.

An alternative method of extracting networks with a higher density in the higher parts of landscape was developed by Tarboton et Ames [35]. Their area threshold value is determined by using the "weighted contributing areas" rather than the normal contributing areas. The weights reflecting local curvature enter into the contributing areas computation so that the initial amount of water is greater in the cells located at the convergent topography. The simplest possibility is to set this amount to two for the cells determined as the river cells by the Peucker&Douglas algorithm and to one for the rest, and then to compute the contributing areas exactly in the same way as described in section 2.2.

**Threshold value determination.** However the type of the threshold was chosen, the proper determination of its value is an additional and a nontrivial question. The simplest method is to visually compare the resulting network with the "blue lines" (e.g. [14], [9], [16]). When the blue lines are converted to an appropriate raster, one can improve such comparison by minimizing the spatial differences between the extracted and the mapped river cells (for such optimization using genetic algorithm see [11]). Montgomery et Foufoula-Georgiou [20] suggested to use the information about the real position of the river sources for the calculation of the critical value of  $aS^2$ .

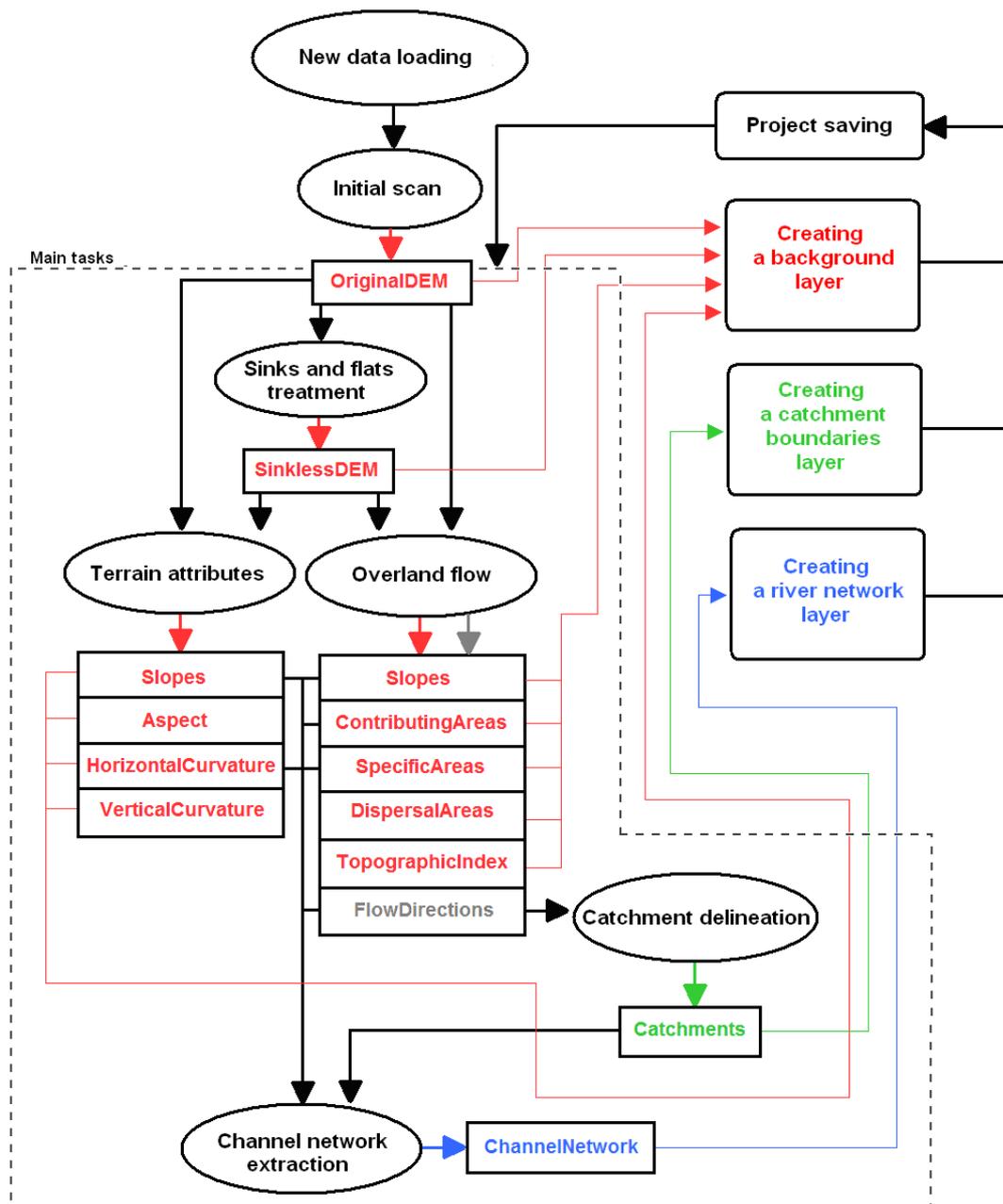
If no such additional information is available, the determination should use some criterion based only on elevations. One way is to analyze the slope-contributing area plot and determine the break points corresponding to river initiations ([33], [8]); however, there is no agreement concerning the proper break point. A completely different approach was developed by Tarboton et al. [33]. It is based on the fulfilment of a few empirical laws (often called Horton's laws) describing the self-similar and scale-invariant morphology of real river networks. One such law is called the "constant drop law" and can be formulated as follows: The average elevation drop along the river reaches with the same Strahler order is approximately constant across the Strahler orders. According to Tarboton et al. (1991), the lowest threshold value leading to a network that satisfies the constant drop law can be considered as the proper one. The authors called this procedure the "constant drop analysis", and an evaluation made by Barták [1] proved its usability.

### 3 Program description

The name of the presented program, DEMETERR 1.1, is an acronym of Digital Elevation Models and their Elementary TERRain analysis. It is created in the Lazarus programming environment using FreePascal language.

### 3.1 Basic structure

The basic structure of the program is shown in Figure 13. The first task that must be done before any other is the Initial Scan, through which the sinks (i.e. the closed depressions) and flats are determined. Then one can perform other tasks, such as Sinks and Flats (treatment of the problematic parts of the DEM), Terrain Attributes (computation of slope, aspect, horizontal and vertical curvature), Overland Flow (the flow directions assignment and contributing areas computation), Topographic Index, Catchment Delineation (using user defined outlet locations or automatically determined outlets on the border of the DEM) and Channel Network (river network extraction). The last can be performed within the whole DEM or within some formerly determined catchment. All implemented extraction methods use some kind of threshold, the setting of which can be done either as a user-defined value or by an automated constant drop analysis. The additional option that can be used is the connecting of discontinuous extracted networks according to flow directions.



**Figure 13.** The basic structure of the program. The tasks are represented as the ovals, the inputs and outputs of the tasks are depicted as the rectangles. The red colour corresponds to background layers, the green colour corresponds to catchment boundaries layer, and the blue colour corresponds to channel network layer.

There is a considerable algorithm choice for each task, covering most of the methods described in literature (see section 3.2 and chapter 2 of this paper). Each task creates some “layers” as outputs that can often enter some other tasks as their inputs. The first layer entering the Initial Scan task is the original DEM. The layers are of the following four basic types:

**Background layers.** Background layers mainly contain the real matrix, whose elements correspond to DEM cells and store a single value for each cell. The typical example is the original DEM, the DEM without sinks and flats, the contributing area layer or the slope layer. Apart from the real matrix they may even contain information about sinks and flats present in the DEM, information about the algorithm through which they were created, used input layers, and some basic descriptive statistics.

**Flow direction layers.** They store the list of all the flow directions of each cell with corresponding weights and information about the method which created them as well as about the input layers.

**Catchment boundaries layers.** They contain information about the position of the outlets and the boundaries of the corresponding catchments as well as about the input layers and method.

**Channel network layers.** They contain a list of the cells belonging to the river network, structured according to Strahler ordering scheme, and, of course, also information about the input layers and the method used.

### 3.2 Implemented algorithms

The overview of the implemented algorithms is given in the following table, with references to the corresponding papers containing the used implementation. Most of the methods use the priority queue data structure and hence have their asymptotic complexity  $O(n \log n)$ , where  $n$  is typically the number of DEM cells; sometimes the complexity is linear (e.g. the contributing areas computation using stack data structure). Such implementations enable the program to process relatively large data.

**Table 1.** Overview of implemented algorithms.

Task	Algorithm	Reference
Closed depressions removal	Filling	[30]
	Outlet Breaching	[17]
	Carving	[31]
	SDI	[36]
	PEM4PIT	[6]
Flat areas treatment	Towards lower gradient	[30]
	Combined gradient	[31]
Slope, aspect, horizontal and vertical curvature	Discrete derivatives	e.g. [4]
	Peucker&Douglas	e.g. [37]
Flow directions assignment	SFD8	[23]
	MFD8	[25]
	SFD $\infty$	[32]
	MFD $\infty$	[28]
River network extraction	Constant area threshold	[14]
	Slope-area threshold	[20]
	Weighted area threshold	[35]
	Relative elevation threshold	[27]
	Constant drop analysis	[33]

### 3.3 Inputs and outputs

**Inputs.** There are three possible ways to approach the work with DEMETERR. The first is to load new data, the second is to open an already existing project, and the third consists in selecting some smaller area from the already loaded DEM and starting a new project with it.

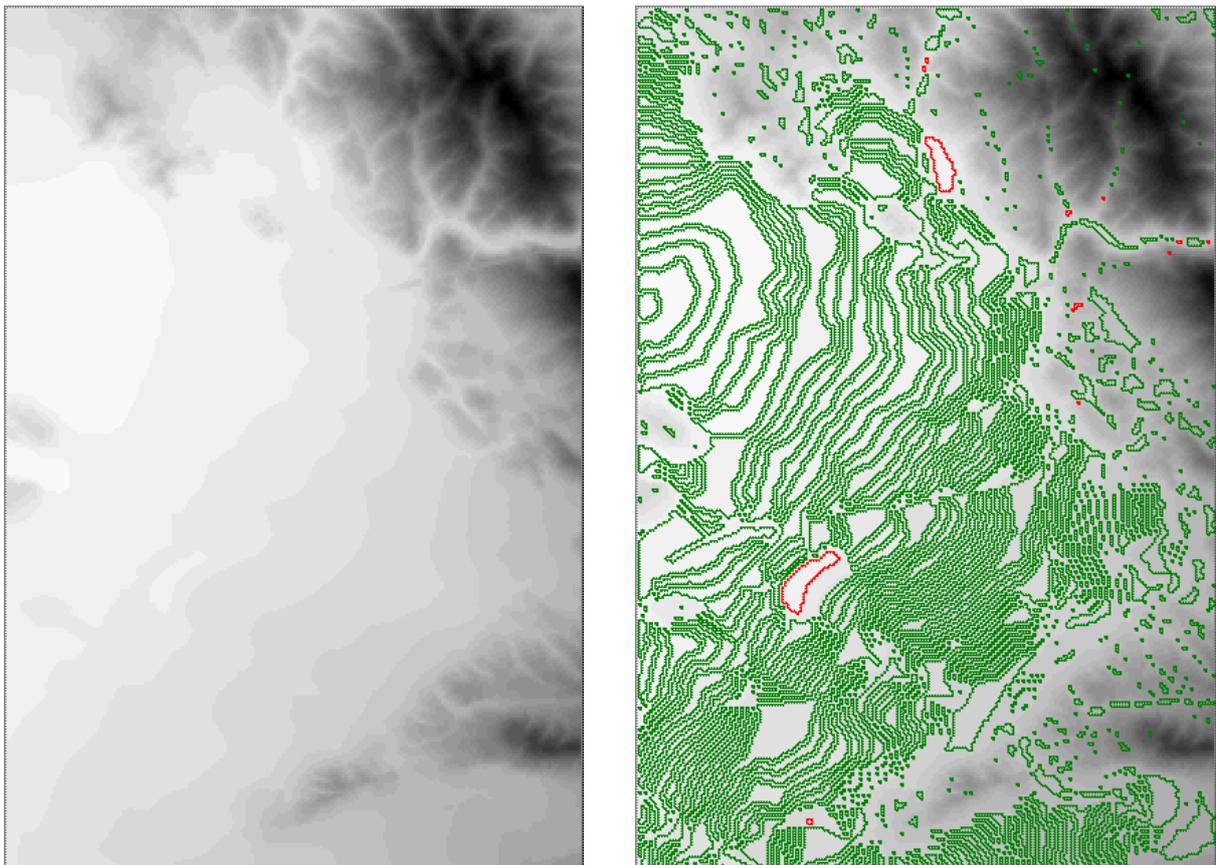
The DEM that has to be loaded must be stored as an ASCII text file, with the same structure as the raster exported from ArcGIS.

**Outputs.** Three types of outputs can be obtained by DEMETERR: the saved project storing all created layers, the ASCII files that can be opened (for example) by ArcGIS, and the graphic outputs in .bmp format.

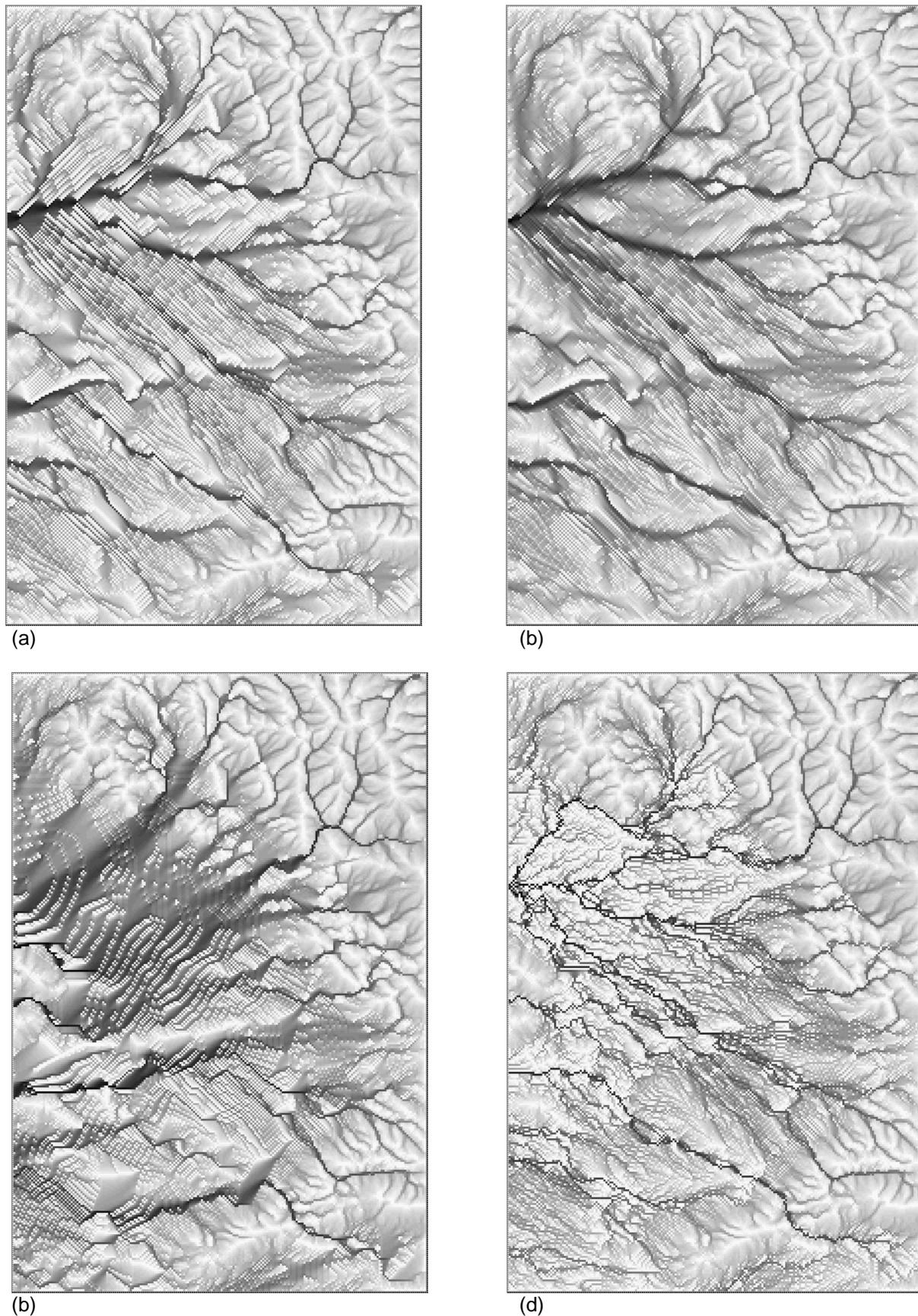
The DEMETERR project consists of the main project file with .dprj extension and a variable number of files without extensions, containing particular layers. In fact, the project file is an ordinary ASCII file storing the information about the original DEM file and the names of all created layers. The layers themselves are also stored in ASCII files and must be saved in the same directory as the project file.

Some layers created in DEMETERR can be exported to ASCII file readable by other programs (e.g. ArcGIS). The resulting raster may be considered continuous (all background layers) or thematic (catchment boundaries and channel network layers), i.e. with the integer values. If the latter is a catchment boundaries layer, the nonzero values belong to cells within any catchment (every catchment has its own value), and all other cells are assigned zero value. In the case of a channel network layer, the zero value represents the non-river cells, and the particular positive integer indicates the Strahler order of the river reach to which the cell belongs.

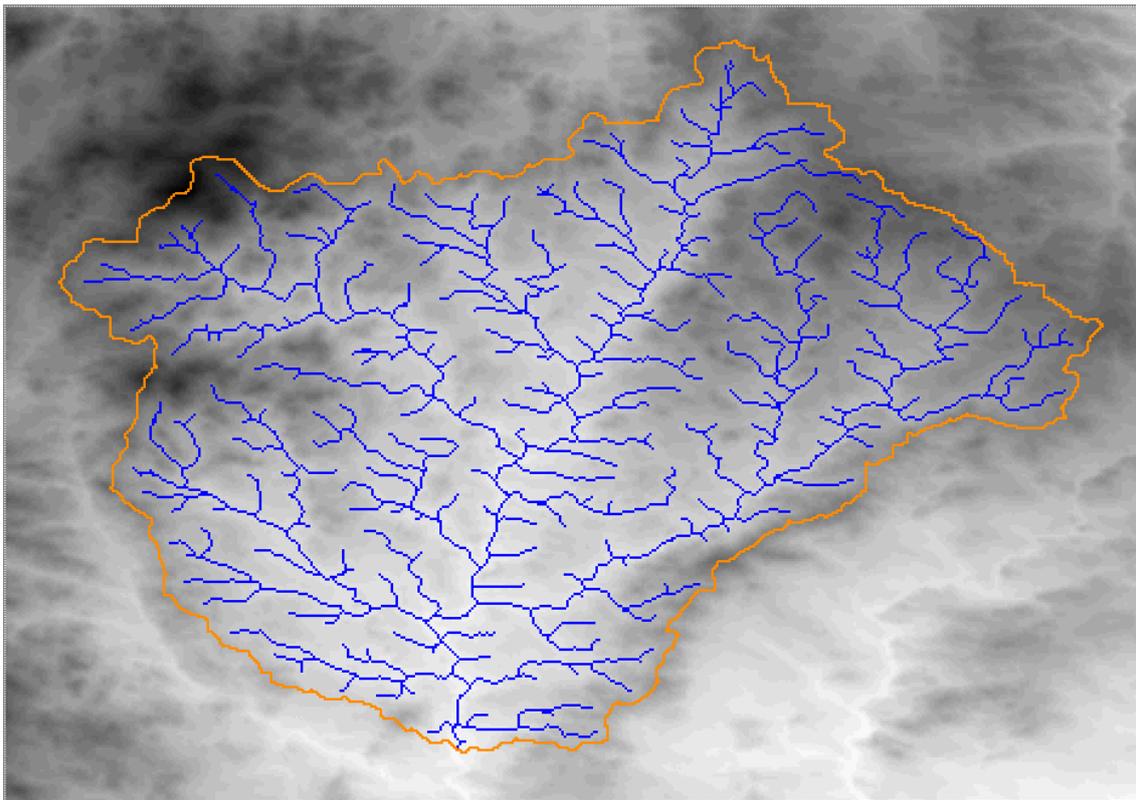
The third type of output is the graphic output. Any layers configuration just displayed on the screen can be saved as a bitmap. Thus the graphical visualization of the task results can be used in any kind of presentation or publication. The examples of graphic outputs are on the Figures 14 to 16, where the USGS DEMs [39], as well as the SRTM DEMs [22] are used.



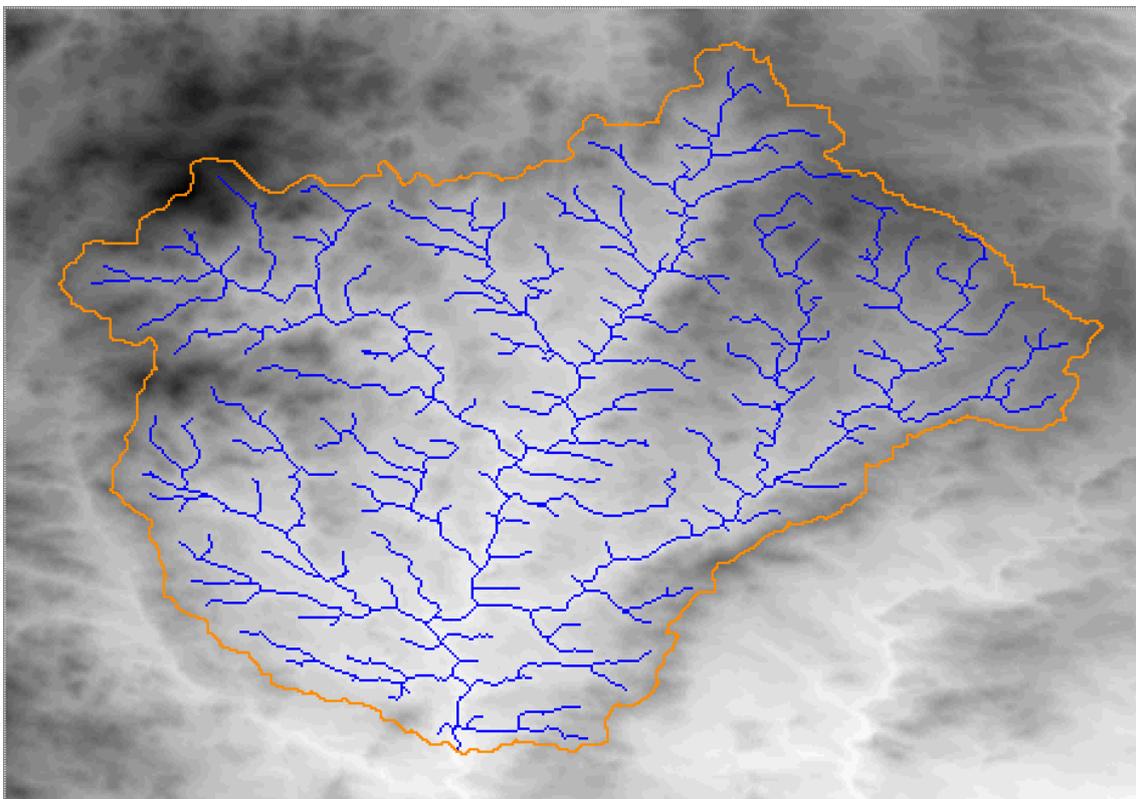
**Figure 14.** The graphic outputs. On the left there is USGS DEM of Altamont, California (USA), on the right there is the same DEM with the depressions (red) and flats (green) shown.



**Figure 15.** The graphic outputs showing the influence of used flats treatment method on the overland flow simulation results (the DEM is from Figure 14). The contributing areas raster created through the MFD8 algorithm are shown. The flats was treated by (a) towards lower gradient, (b) combined gradient, (c) SDI, and (d) PEM4PIT (see section 2.5 of this paper).



(a)

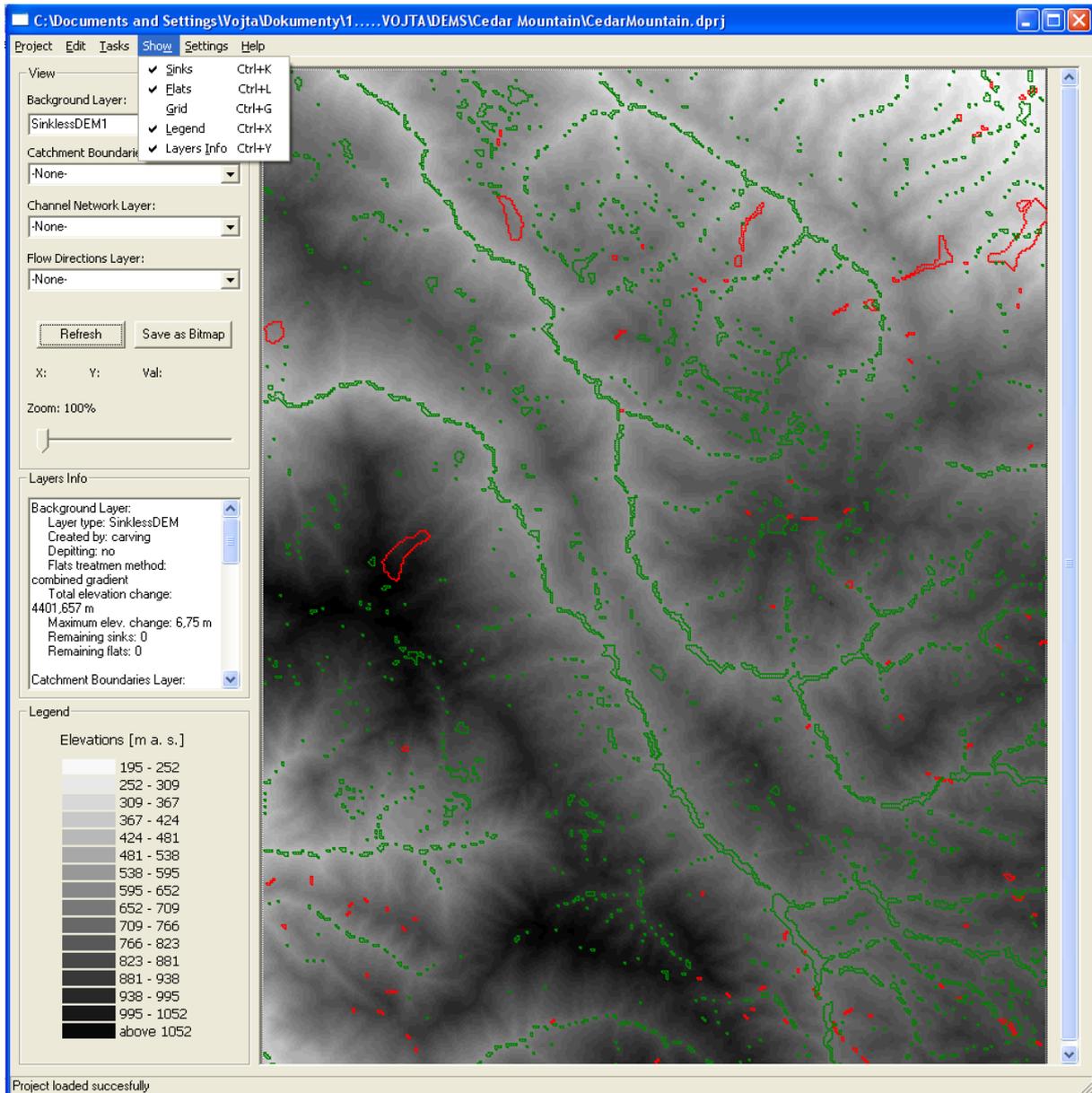


(b)

**Figure 16.** The graphic outputs showing two river networks extracted from the Moravská Dyje SRTM DEM, through the constant area threshold (a), and weighted area threshold (b) methods. Both networks are extracted using the constant drop analysis (see section 2.6 of this paper), the latter is quite similar to the “blue lines” drawn in the topographic map of scale 1:25 000 [1].

### 3.4 Graphical user interface

The graphical user interface is user-friendly, using standard Windows dialogs (see Figures 17 to 19). The main window serves as the environment for calling particular tasks and viewing the results. Each task has its own dialog with algorithm choice and parameters setting. Maximum four layers can be displayed simultaneously; one background layer, one flow direction layer, one catchment boundaries layer and one channel network layer. Together with each layers configuration, information about the layers can be displayed, as well as the sinks and flats. The picture can be zoomed in, and at a sufficient resolution the flow directions can be drawn as single arrows directed along the steepest slope. All colours and line widths can be set by the user in the Drawing Options dialog.



**Figure 17.** The graphical user interface. The raster is the USGS DEM of Cedar Mountains, California (USA). The depressions are depicted in red, the flats in green.

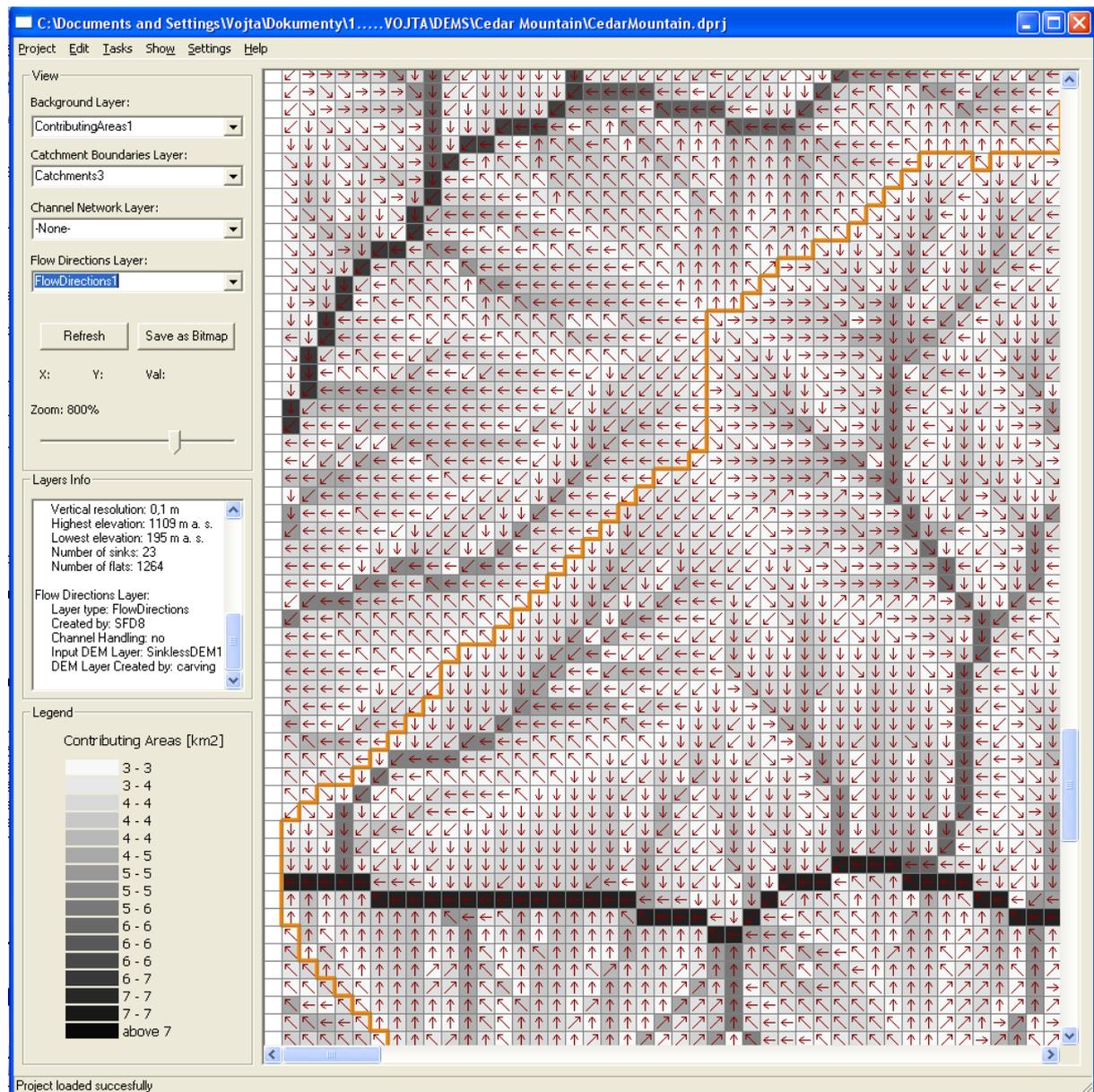
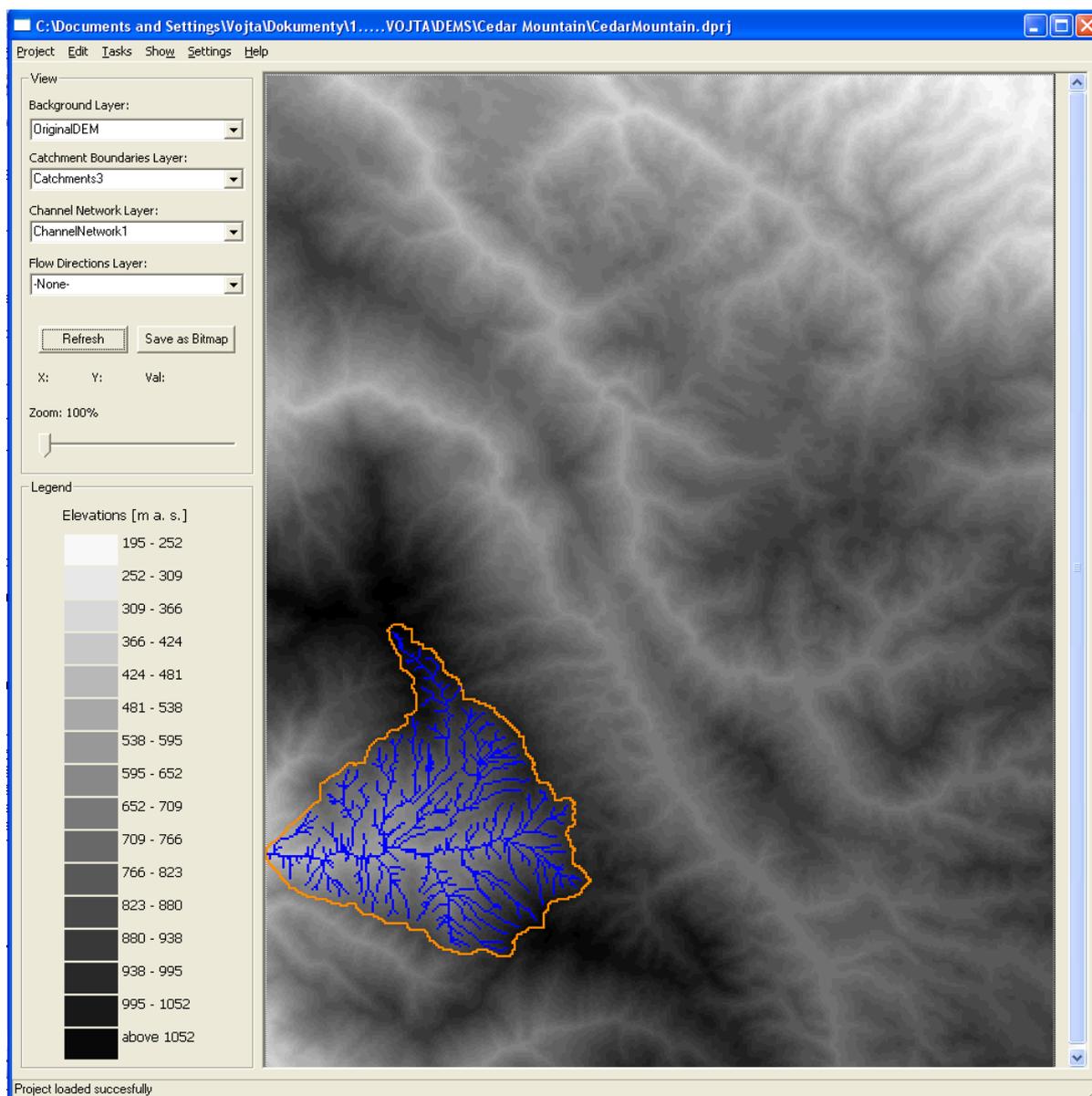


Figure 18. The graphical user interface. The DEM is zoomed in so that the flow directions are drawn.

#### 4 Accessibility and further development

The licence agreement, as well as the program itself, is available on the <http://fzp.czu.cz/~bartakv/demeter>. Free use of the program is restricted to non-commercial (i.e. scientific and educational) purposes and is conditioned by the author's pronounced permission.

Further development of the presented program will be focused on the improvement of the input/output options in order to increase the compatibility with other, commonly used, programs, such as ArcGIS or GRASS. Thus in further versions, the input file format choice should appear, as well as the possibility of exporting layers to standard raster formats.



**Figure 19.** The graphical user interface. The channel network layer and the catchment boundaries layer are displayed over the background layer (DEM).

## References

1. Barták V. Algoritmy pro zpracování digitálních modelů terénu s aplikacemi v hydrologickém modelování. Diploma thesis. Czech University of Life Sciences, Prague 2008. URL: <http://fzp.czu.cz/~bartakv/thesis> [cit. 2009-11-23].
2. Florinsky I.V. Errors of signal processing in digital terrain modelling. *International Journal of Geographical Information Science* **16**(5). 2002. ISSN 1362-3087.
3. Freeman T.G. Calculating catchment area with divergent flow based on regular grid. *Computers&Geosciences* **17**(3). 1991. ISSN 0098-3004.
4. Gallant J.C., Wilson J.P. TAPES-G: a grid-based terrain analysis program for the environmental sciences. *Computers&Geosciences* **22**(7). 1996. ISSN 0098-3004.
5. Garbrecht J., Martz L.W. The assignment of drainage direction over flat surfaces in raster digital elevation models. *Journal of Hydrology* **193**(1-4). Amsterdam 1997. ISSN 0022-1694.

6. Grimaldi S., Nardi F., Di Benedetto F., Istanbuluoglu E., Bras R.L. A physically-based method for removing pits in digital elevation models. *Advances in Water Resources* **30**(10). 2007. **ISSN** 0309-1708.
7. Holmgren P. Multiple flow direction algorithms for runoff modelling in grid based elevation models: an empirical evaluation. *Hydrological Processes* **8**(4). Chichester 1994. **ISSN** 0885-6087.
8. Ijjasz-Vasquez E.J., Bras R.L. Scaling regimes of local slope versus contributing area in digital elevation models. *Geomorphology* **12**(7). 1995. **ISSN**: 0169-555X.
9. Jenson S.K., Domingue J.O. Extracting topographic structure from digital elevation data for geographic information system analysis. *Photogrammetric Engineering and Remote Sensing* **54**(11). 1988. **ISSN** 0099-1112.
10. Jones R. Algorithms for using a DEM for mapping catchment areas of stream sediment samples. *Computers&Geosciences* **24**(4). 2002. **ISSN** 0098-3004.
11. Kim S., Lee H. A digital elevation analysis: a spatially distributed flow apportioning algorithm. *Hydrological Processes* **18**(10). Chichester 2004. **ISSN** 0885-6087.
12. Lindsay J.B., Creed I.F. Removal of artifact depressions from digital elevation models: towards a minimum impact approach. *Hydrological Processes* **19**(16). Chichester 2005. **ISSN** 0885-6087.
13. Lindsay J.B., Creed I.F. Distinguishing actual and artefact depressions in digital elevation data. *Computers&Geosciences* **32**(8). 2006. **ISSN** 0098-3004.
14. Mark D.M. Automated detection of drainage networks from digital elevation models. *Cartographica* **21**(2-3). Toronto 1984. **ISSN** 0317-7173.
15. Martz L.W., De Jong E. CATCH: a FORTRAN program for measuring catchment area from digital elevation models. *Computers&Geosciences* **14**(5). 1988. **ISSN** 0098-3004.
16. Martz L.W., Garbrecht J. Numerical definition of drainage network and subcatchment areas from digital elevation models. *Computers&Geosciences* **18**(6). 1992.. **ISSN** 0098-3004.
17. Martz L.W., Garbrecht T J. An outlet breaching algorithm for treatment of closed depressions in a raster DEM. *Computers&Geosciences* **25**. 1999. **ISSN** 0098-3004.
18. Meisels A., Raizman S., Karnieli A. Skeletonizing a DEM into a drainage network. *Computers&Geosciences* **21**(1). 1995. **ISSN** 0098-3004.
19. Montgomery D.R., Dietrich W.E. Landscape dissection and drainage areaslope thresholds. In: Kirkby M.J. (ed.) *Process models and theoretical geomorphology*. Wiley, New York 1994. **ISBN** 0471-94104-2.
20. Montgomery D.R., Foufoula-Georgiou E. Channel network source representation using digital elevation models. *Water Resources Research* **29**(12). Washington 1993. **ISSN** 0043-1397.
21. Moore I.D., Grayson R.B., Ladson A.R. Digital terrain modelling: a review of hydrological, geomorphological, and biological applications. *Hydrological Processes* **5**(1). Chichester 1991. **ISSN** 0885-6087.
22. NASA. Shuttle Radar Topographic Mission – the mission to map the world. URL: <http://www2.jpl.nasa.gov/srtm/> [cit. 2008-08-08].
23. O'Callaghan J.F., Mark D.M. The extraction of drainage networks from digital elevation data. *Computer Vision, Graphics, and Image Processing* **28**(3). San Diego 1984. **ISSN** 0734-189X.
24. Quinn P., Beven K., Chevallier P., Planchon O. The prediction of hillslope flow paths for distributed hydrological modelling using digital terrain models. *Hydrological Processes* **5**(1). Chichester 1995. **ISSN** 0885-6087.
25. Quinn P.F., Beven K.J., Lamb R. The  $\ln(a/\tan\beta)$  index: how to calculate and how to use it within the TOPMODEL framework. *Hydrological Processes* **9**(2). Chichester 1995. **ISSN** 0885-6087.
26. Rieger W. (1998) A phenomenon-based approach to upslope contributing area and depressions in DEMs. *Hydrological Processes* **12**(6). Chichester 1998. **ISSN** 0885-6087.
27. Roth G., La Barbera P., Greco M. (1996) On the description of the basin effective drainage structure. *Journal of Hydrology* **187**. Amsterdam 1996. **ISSN** 0022-1694.
28. Seibert J., McGlynn B.L. New triangular multiple flow direction algorithm for computing upslope areas from gridded digital elevation models. *Water Resources Research* **43**. Washington 2007. **ISSN** 0043-1397 .
29. Soille P. Optimal removal of spurious pits in digital elevation models. *Water Resources Research* **40**. Washington 2004. **ISSN** 0043-1397.
30. Soille P., Gratin Ch. An efficient algorithm for drainage network extraction on DEMs. *Journal of Visual Communication and Image Representation* **5**(2). 1994.

- 
31. Soille P., Vogt J., Colombo R. Carving and adaptive drainage enforcement of grid digital elevation models. *Water Resources Research* **39**. Washington 2003. **ISSN** 0043-1397.
  32. Tarboton D.G. A new method for determination of flow directions and upslope areas in grid digital elevation models. *Water Resources Research* **33**(2). Washington 1997. **ISSN** 0043-1397.
  33. Tarboton D.G., Bras R.L., Rodriguez-Iturbe I. On the extraction of channel networks from digital elevation data. *Hydrological Processes* **5**(1). Chichester 1991. **ISSN** 0885-6087.
  34. Tarboton D.G., Bras R.L., Rodriguez-Iturbe I. A physical basis for drainage density. *Geomorphology* **5**(59-76). Amsterdam 1992. **ISSN** 0169-555X.
  35. Tarboton D.G., Ames D.P. Advances in the mapping of flow networks from digital elevation data. *World Water and Environmental Resources Congress*. Orlando, Florida 2001. URL: <http://www.neng.usu.edu/cee/faculty/dtarb/tarpubs.htm> [cit. 2008-04-05].
  36. Tianqi A., Takeuchi K., Ishidaira H., Yoshitani J., Fukami K. (2003) Development and application of a new algorithm for automated pit removal for grid DEMs. *Hydrological Sciences Journal* 48(6): 985-997. Wallingford 2003. **ISSN** 0262-6667.
  37. Tribe A. Automated recognition of valley lines and drainage networks from grid digital elevation models: a review and a new method. *Journal of Hydrology* 139(1-4). Amsterdam 1992. **ISSN** 0022-1694.
  38. Turcotte R., Fortin J.-P., Rousseau A.N., Massicotte S., Villeneuve J.-P. Determination of the drainage structure of a watershed using a digital elevation model and a digital river and lake network. *Journal of Hydrology* **240**. Amsterdam 2001. **ISSN** 0022-1694.
  39. U. S. Geological Survey. Digital elevation model (DEM). URL: <http://eros.usgs.gov/#/Guides/dem> [cit. 2008-08-08].
  40. Wang L., Liu H. An efficient method for identifying and filling surface depressions in digital elevation models for hydrologic analysis and modelling. *International Journal of Geographical Information Science* **20**(2). Online 2006. **ISSN**: 1362-3087.