Theory and Technology in Natural Sciences and Watershed Management

Earth Systems Institute

Programs for DEM Analysis

by

Daniel Miller

2002

Software and documentation by Daniel Miller
Direct inquiries to **danmiller@earthsystems.net**

These programs are a direct outgrowth of work sponsored by the Coastal Landscape Analysis and Modeling Study (http://www.fsl.orst.edu/clams/). Support has been provided by the US Forest Service, the US Bureau of Land Management, the National Marine Fisheries Service, and the Oregon Department of Geology and Mineral Industries. Lee Benda, with Earth Systems Institute; Kelly Burnett and Kelly Christiansen, with the USDA Forest Service, Pacific Northwest Research Station; Sharon Clarke, at the Department of Forest Sciences, Oregon State University; Jon Hofmeister, with the Oregon Department of Geology and Mineral Industries; and Mindy Sheer, with the National Marine Fisheries Service, Northwest Fisheries Science Center, have all provided assistance in development and testing.

Earth Systems Institute is a nonprofit organization committed to interdisciplinary research in landscape interactions and applications for resource management. We explore interactions between climate, vegetation, geomorphology, hydrology, and aquatic ecology, focusing on climatic disturbances, erosion, terrestrial-channel linkages, biodiversity, and effects of land use. We transfer new field methods, models, and theories into applications for natural resource management.

Table of Contents

INTRODUCTION

Numerical simulation models play an important role in efforts to explore watershed interactions. Such models require quantification of a variety of topographic attributes, hence, analyses of digital elevation data forms an important component of our modeling efforts at ESI. We've focused on use of point elevation values provided over a regular grid, typically referred to as digital elevation models (DEMs), since data in this format are widely available. Topographic analyses prove useful for a variety of efforts beyond simulation modeling, so we've provided capabilities for output of derived quantities in raster and vector formats accessible to geographic information systems (GIS). This document provides instructions for use of these programs and descriptions of the file formats and algorithms used.

These programs form part of an active research program and undergo continuous development and testing. Updates and additions to the programs are posted to the ESI website at www.earthsystems.net.

What the Programs Do

Using DEMs, a set of user-specified parameters, and other optional data sources (e.g., vegetation cover), these programs will

- delineate a routed channel network, with an option for drainage enforcement,
- estimate certain channel and valley attributes (drainage area, channel gradient, channel length, mean annual precipitation, valley width, valley side-slope gradient),
- estimate susceptibility to shallow colluvial landsliding using either surface gradient or the SHALSTAB algorithm,
- estimate potential for debris flow scour and deposition in channels, and calculate a relative volume of debris-flow-derived sediment for each channel reach, and
- estimate debris flow inundation zones.

Channel attributes and debris-flow-scour and deposition probabilities are averaged over reaches and output to an ArcInfo Shape file. Information about tributary junctions (e.g., junction angle, tributary drainage area) may be output to a separate (point) Shape file. (Information on ArcInfo shape files is available at http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf). Landslide susceptibility and debris-flow-inundation-hazard zones are output to digital or ASCII raster files.

The size of the DEM that can be examined with these programs is set in part by the types of analyses you require and by the available memory and disk space on your computer. Any debris-flow analysis adds considerably to the memory requirements and to the run time. The programs have been used successfully on DEMs containing ~70 million pixels; data sets larger than this cannot currently be accommodated.

The programs use a lot of disk space for intermediate storage of results. Depending on the analyses specified, the programs may require work space on the disk in excess of 10 times the size of the DEM.

Command-Line Implementation

We provide no graphical interface for these programs. Rather, they are run from command lines and use ASCII files for specification of adjustable parameters. This allows use of the same source code for different platforms; the programs can be compiled for a Windows machine or for a Unix machine and the implementation will be the same on both.

Caveats

You are responsible for how results from these programs are used. The accuracy of derived quantities is a function of the accuracy and resolution of input data and of the ability of the algorithms (and their implementation) to use that data to infer other quantities that have not been directly measured. Additionally, despite my utmost diligence, there may be undetected bugs yet lurking both in the code and in the logic employed in implementation of the algorithms. You must assess the appropriateness of the algorithms for your application, my success at implementing them (source code is available), and verify the accuracy of the results.

USING THE PROGRAMS

Program Files

The programs are divided into two primary modules, Bld_grds and Netrace, and a set of five utilities. Each is described briefly here, with details provided in subsequent sections.

Bld_grds primary tasks are to determine the flow direction and contributing area for every DEM pixel. The flow direction specifies the outflow direction for surface water exiting the pixel and the contributing area specifies the surface area draining to the pixel, i.e., the pixel "watershed". In the process of calculating these quantities, Bld_grds produces auxiliary files specifying surface gradient, contour length per pixel, the number of adjacent inflowing pixels. Bld_grds can also use a separate channel mask for drainage enforcement, derived from an existing line coverage of channel locations, for example, to aid in estimating flow directions (and channel locations) through areas of flat topography.

Netrace uses output files created by Bld_grds, along with the DEM, to trace a "routed" channel network and to estimate topographically controlled channel and valley attributes. In a "routed" network, flow directions and tributary connections are all determined so that flow, for example, can be followed (routed) up or downstream through the network. Netrace can continue to trace flow paths up hillslopes to pixels identified as landslide prone. These flow paths can then be evaluated in terms of their potential for delivery of sediment to channels by landslides and debris flows.

Adjustable parameters shared by the two programs are specified in the ASCII file parameters.dat. Parameters used only by Netrace for calculating debris flow inundation hazards are specified in the ASCII file runout_parameters.dat. Output options are specified in the ASCII file instructions.txt. By default, these files must be placed in a directory on the c: drive named \work\source\.

Both programs use a digital raster file format the same as that used in the GIS IDRISI (http://www.clarklabs.org). Three utilities are provided for file conversions:

Readdem – a program to read a USGS-format DEM and produce a digital raster file for Bld_grds.

Readgrd – a program to read an ArcInfo-format ASCII raster file and produce a digital raster file for Bld_grds.

GridASCII – a program to read a digital raster file and produce an ArcInfo-format ASCII raster file.

Two other utilities are also provides:

Merge – a program to merge multiple digital raster DEMs into a single file. This program will fill in the single-pixel gaps that sometimes occur, but currently performs no smoothing at DEM boundaries.

Splice – a program to merge multiple shape files produced by Netrace and update all attributes.

Data Files

All files must be located in the same directory. When running the programs, you'll be prompted for the "root directory", indicating the directory containing the data files.

All raster data files use a specific naming convention that includes an ASCII identification code that may be up to 5 characters long. For example, elevation data (the DEM) is named elev_ID.ext, where "ID" represents the (up to) 5 ASCII characters used for the identification code and "ext" refers to the file extension. All files associated with this DEM will use the same ID. When running the program, you'll be prompted for the ASCII ID.

A DEM alone is sufficient to run these programs. They can also use certain other types of information, if data are available, described below.

Elevation – (elev_ID, required). Digital elevation data, used by both Bld_grds and Netrace.

Mean Annual Precipitation – (prec_ID, optional) Bld_grds will read a raster file providing mean annual precipitation depth for each DEM pixel and use this to determine the mean annual precipitation volume for the contributing area of each pixel. This information is then used by Netrace to determine the mean annual precipitation depth for the drainage area to each channel reach.

Vegetation Cover – (veg_ID, optional) Netrace can use a grid of vegetation classes in determining landslide susceptibility and debris-flow-runout length. The program currently recognizes four classes: 1) open stands (clear cuts); 2) mixed hardwood – conifer stands and small conifer stands; 3) large conifer stands; and 4) roads (with a buffer). The effects of vegetation cover are currently "hard-wired" into the program, but will be offered as adjustable parameters in future additions.

Channel Mask – (chan_ID, optional) Bld_grds will read a raster file that flags pixels containing a channel according to a separate data source (e.g., a vector channel coverage). Channel locations determined by Netrace are based on flow directions estimated from the DEM. In many cases the DEM provides insufficient information to determine a unique flow direction (e.g., in topographically flat areas), or there may be errors in the DEM that result in incorrect channel locations. The channel mask incorporates additional information in determination of flow directions and channel locations and serves to provide a certain level of drainage enforcement. The degree to which the channel mask is enforced is determined by the value of the flagged cells. This value specifies the maximum depth of incision (in integer decimeters) used to direct flow directions; the greater its value, the greater is the influence of the channel mask. This value may vary over the extent of the channel mask, thus allowing the user to increase the level of drainage enforcement over certain portions of the DEM. If a negative value is used to flag channel pixels, then a constant incision depth is used, equal to the value specified by the parameter "dig" in the file parameters.dat.

Lake Mask – (lake_ID, optional) Netrace cannot identify lakes from the DEM, yet it is useful to flag "channel reaches" that actually cross bodies of water. The channel mask is a raster grid that identifies DEM pixels located within lakes and reservoirs; all pixels have value zero, except those falling on a body of water as identified with a separate GIS coverage. These pixels are assigned the integer identifier of that body of water, which is then assigned to the corresponding channel reach.

Two data formats are currently supported: the IDRISI binary raster format and the Band Interleaved by Line format.

IDRISI binary format: Requires an ASCII file with metadata (extension .rdc) and a binary file of cell values (extension .rst). The elevation file may be either integer or real. For typical 10-meter DEMs with vertical resolution of 0.1 meter, the elevation file will be real. The channel, lake, and precipitation files are integer. Binary files consist of a string of values (4-bytes for real numbers, 2-bytes for integer) ordered by row. An ASCII metadata file (e.g., elev_1.rdc) is reproduced below:

```
file format : IDRISI Raster A.1
file title  :
data type   : real
file type   : binary
columns     : 991
rows        : 1401
ref. System : utm10n
ref. Units  : m
unit dist.  : 1.0000000
min. X      : 568220.0000000
max. X      : 578120.0000000
min. Y      : 5038660.0000000
max. Y      : 5052660.0000000
pos'n error : unknown
resolution  : 10.0000000
min. value  : 0.4572000
max. value  : 123.6573639
display min : 0.4572000
display max : 123.6573639
value units : meters
value error : unknown
flag value  : -9999.0000000
flag def'n  : nodata
```

BIL files (Band Interleaved by Line) files: These are integer files and therefore imported elevation data (if given in units of meters, for which vertical resolution is commonly 0.1 meter) must be in units of decimeters (0.1 meter). Bld_grds will then convert to meters internally. BIL files require three separate files: a binary file containing cell values, with extension .bil; an ASCII header file, with extension .hdr; and an ASCII file providing corner coordinates and DEM horizontal resolution, with extension .blw.

Order of Operation

1) Input data files must be written to an appropriate format and given the proper names. The names are specified above. Currently the formats that work are the IDRISI binary raster format (using extensions ".rst" and ".rdc" for the binary file and metadata, respectively) and the Band Interleaved by Line format (using extensions ".bil" for the binary file, ".hdr" for the ASCII header, and ".blw" for geographic locational information). The utility Readdem may be used to convert an ASCII USGS-format DEM to the IDRISI format; the utility Readgrd may be used to convert an ArcInfo-format ASCII grid to IDRISI format. Make sure that DEMs use the same units for horizontal and vertical distances. Some USGS DEMs come with vertical units of feet and horizontal units of meters; you'll have to convert the vertical units to meters prior to using these DEMs. Likewise, some DEMs use vertical units of decimeters and horizontal units of meters. These, too, must have the vertical units converted to meters prior to using them with Bld_grds.
2) Set all parameter values in the ASCII file "parameters.dat" and, if you are including debris-flow-inundation hazards, in "runout_parameters.dat". Make sure that these files are placed on the c: drive in the directory \work\source\, as this is where Bld_grds and Netrace will look for them. Currently, if debris-flow probabilities are to be calculated, the file dnda.dat also needs to be placed in the c:\work\source\ directory. This file provides a relationship between a topographic index of slope stability and landslide density calibrated for the Oregon Coast Range.
3) Process the DEM (and other optional input files) using Bld_grds to create topographic outputs. This step will produce several binary raster files on your hard disk.
4) Specify the analyses required for Netrace in the ASCII file "instructions.txt".
5) Run Netrace to create the output Shape files.
6) Use Splice to merge Shape files, if needed.
7) Delete unneeded raster files.

Batch files can be created to automate these tasks, an example of which is provided in a later section.

UTILITIES

Readdem

This program reads an ASCII USGS-format DEM and creates a binary IDRISI-format raster file, consisting of a binary raster file with extension ".rst" and an ASCII metadata file with extension ".rdc" (described previously).

The program is initiated by typing “readdem” at a command-line prompt from the directory containing the executable file. It will prompt you for the “Root directory:”, for which you specify the location of the DEM file, including the full path (with the final “\” for Windows machines or “/” for Unix machines). You’ll then be prompted for the DEM-file name (including any extension). After reading the file, you’ll be prompted for an output file name. If the file is intended for use with Bld_grds, name it elev_ID, where “ID” indicates an ASCII identifier up to five characters long. You’ll also be asked for the “Quadrangle name”, referring to the name of the USGS 7.5-minute quadrangle the DEM corresponds to. This name will be included in the metadata.

Readgrd

Readgrd will read an ASCII raster file created by ArcInfo and create an IDRISI-format raster file. The program is initiated by typing “readgrd” at a command line from the directory containing the executable file. You’ll then be prompted for the “root directory:” and for the name of the ASCII grid, which requires the extension “.grd”. So, if you have a an ASCII grid file named dem1.grd in directory c:\Washington\dems\, you’d type “c:\washington\dems\” after the “root directory:” prompt, and you’d type “dem1” when asked for the name of the file.

You’ll then be asked for an output file name, which if you’re planning on using the file with Bld_grds, should be elev_ID, where ID is an ASCII identifier up to five characters long. The next prompt asks if the file is integer or real. You’ll then be asked to specify a “Grid multiplication factor:”. This allows you to convert vertical units in the output file; for example, if the DEM uses vertical units of decimeters and horizontal units of meters, specify a multiplication factor of 10.0. If the units are the same, simply specify 1.0. The last prompt asks for the DEM unit of measure, either meters or feet.

GridASCII

GridASCII will create an ASCII file using the ArcInfo raster format. Invoke the program by typing “gridASCII” at a command prompt in the directory containing the executable file. You’ll then be asked for the “root directory:” and for a list of file name. Type file names with no extension. A blank return indicates the end of the list.

Merge

Merge will combine a set of DEMs (in IDRISI format) into a single file. You must specify the root directory and then a list of file names (with no extension). The program will check that all files use the same units and horizontal resolution and will fill in single-cell gaps that sometimes occur between DEMs. You also have the option of creating an ASCII output file.

Splice

Splice will take a set of shape files produced by Netrace and combine them into a single file with corrected reach attributes (e.g., drainage area). With this facility, a large DEM may be divided into a set of smaller tiles, shape files produced for each, and then the shape files merged to provide a single file for an entire large drainage, with the correct LLIDs, drainage areas, mean annual precipitation values, and channel lengths carried through. Note that there MUST be overlap between the shape files (and, therefore, between the DEM tiles), otherwise splice has no way of knowing how the shape files are connected.

Splice will prompt you for the number of shape files to merge and then for the name of each, including the full path. This allows you to merge shape files held in different directories. You'll then be asked for the name of the output shape file, including the full path.

BLD_GRDS

Command-Line Prompts

Bld_grds is invoked by typing "Bld_grds" at a command line from the directory holding the executable file. It asks for the "Root Directory:" (the path to the data files) and for the file identifier (an ASCII sequence from one to five characters long).

It then asks if "Zero values indicate nodata (y/n):" Some DEMs have used a value of zero over areas covered by ocean. Since zero values are sometimes legitimate elevations in low-lying coastal areas, it is impossible to distinguish between oceanic and terrestrial portions of the DEM. Bld_grds will attempt to define a flow direction for every pixel with data in the DEM. Doing so over large portions of ocean is both time consuming and meaningless, and in some instances will cause the program to crash as it runs out of stack space in memory. Hence, you'll see the question posed at the beginning of the program run "Zero values indicate nodata (y/n):". If you indicate "y", any pixels with an elevation value of zero will be ignored.

Bld_grds then reads the data files and asks if you want to do slope stability calculations, i.e., "Do slope stability (y/n):". If you respond "y", an additional raster grid will be created holding values corresponding to the steady-state precipitation intensity required for failure calculated using the SHALSTAB algorithm. Note that my implementation of this algorithm is slightly different than that provided by the Berkeley group (http://socrates.berkeley.edu/~geomorph/).

If Bld_grds finds a file with the name ang_ID, it will ask if you want to use the existing flow direction file. If you don't, simply type "n".

Outputs

Bld_grds creates a series of raster files, all in IDRISI binary format. These are:

Accum_ID: A flow accumulation grid, giving an estimate of the drainage area to each pixel. Drainage area in accum_ID is given in units of DEM pixels. Negative values indicate edge contamination, i.e., potentially underestimated drainage area because the watershed intersected the DEM boundary.

Slope_ID: Estimated surface gradient at each DEM point.

Bcont_ID: Estimated total contour length crossed by flow into each pixel, in units of DEM cells. Values can vary from 0 to 4. Low values of Bcont indicate local flow convergence; high values indicate local flow divergence. Note that accum/bcont gives specific drainage area (drainage area per unit length of contour).

Pin_ID: The area of the eight adjacent cells flowing into each pixel. Values vary from 0 to 8. Pin provides an estimate of local flow convergence.

Ang_ID: Flow direction for each pixel, calculated using the algorithm described by Tarboton (1997). Values vary from 0 to 2*PI radians.

Dir_ID: D8 flow direction for each pixel, with flow directions confined to one of the eight adjacent pixels (i.e., N, NE, E, SE, S, SW, W, NW).

Pval_ID: Mean annual precipitation value for the drainage area to each pixel.

Algorithms

The first step is to read the input grids. The elevation grid is loaded into memory and zero-shifted (the minimum elevation is subtracted from all values). The program checks to see if the channel mask and the precipitation grids are present in the data directory. Subsequent tasks are described below. Alterations to the DEM are stored in a temporary raster file, named etmp_ID, which is deleted upon successful completion of the program.

Creation of a Depression-Less DEM

It is assumed that the DEM contains no closed depressions – that all pixels eventually drain to a DEM edge. However, most DEMs contain closed depressions that must somehow be removed so that drainage directions can be defined. These depressions arise both from actual errors in the DEM (e.g., poor edge matching between USGS quadrangles) and because pour points may be missed by the point samples of elevation. I use two algorithms to remove closed depressions: filling of depressions to the level of the lowest pour point and incision of probable channel locations to drain closed depressions.

<u>Filling</u>: This is the standard algorithm, described by Jenson and Domingue (1988): find the lowest pour point and raise all pixels in the depression to this level. Unfortunately, this procedure tends to obscure potentially useful information that elevations within the depression may actually provide. Hence, I first try another strategy, cutting of channels along the lowest pour point.

<u>Cutting</u>: I find the lowest pour point and then try to infer drainage away from the depression by tracing a path that proceeds to the lowest adjacent pixel. If a drainage path can be found with a length of five pixels or less, and that requires excavation of no more than 10 DEM units (typically meters), I'll lower the pixel elevations along the drainage path to the level of the pour point. If a drainage path meeting these requirements cannot be found, the depression is filled to the level of the pour point.

Flow Directions

A flow direction must be defined for every pixel within the DEM. Where elevation differences between adjacent pixels allow determination of a flow direction, I use an algorithm presented by Tarboton, (1997). This algorithm represents the ground surface local to the DEM point in terms of 8 triangular facets, with corner elevations defined by the DEM point and its eight adjacent points. Flow directions are defined for each facet; the steepest outgoing flow path is then assigned to the pixel. Note that this algorithm defines flow directions that may be in any direction and are not limited to one of the eight directions directed to an adjacent DEM point. Flow direction values, in radians of azimuth from north, are stored in raster file ang_ID.

Flow directions determined as described above allow flow into, at most, two downslope pixels, thus allowing flow dispersion over topographically divergent areas. Once the criterion for channel initiation has been met, dispersion is no longer allowed and alternative algorithms are used to determine flow direction into one of the adjacent pixels, as discussed below.

Two other quantities are derived with the calculation of flow direction: 1) a measure of local flow convergence, specified by the area (in terms of DEM pixels) of the eight adjacent pixels that flow into each pixel, stored in raster file pin_ID, and 2) a measure of the contour length traversed by incoming flow to each pixel, stored in raster file bcont_ID. I'll discuss this quantity at greater length below.

For pixels that have no adjacent pixels of lower elevation, a flow direction is undefined. These are flat areas within the DEM. To define flow directions over flat areas I use an algorithm described by Garbrecht and Martz (1997), which tends to direct flow away from adjacent higher elevations and towards adjacent lower elevations. Flow directions within flats are then directed toward one of the eight adjacent pixels.

A D8-flow direction grid (with flow direction an integer multiple of PI/4 radians, so that all flow is directed towards only one adjacent pixel) is also created during the calculation of flow directions. This grid is named dir_ID and is used for routing of debris flows, for which downslope dispersion is also not allowed. For pixels meeting the criteria for channels, the D8 flow direction will match that written to ang_ID. For all other pixels, the flow direction specified in dir_ID and ang_ID may diverge slightly. To compensate for the limited number of flow directions allowed with the D8 method, there is an option to allow a random element into determination of D8 flow direction. I use the partitioning of flow indicated by the flow direction (Tarboton 1997) specified in ang_ID as a measure of the probability that flow (or a debris flow) will enter each adjacent downstream pixel. Sampling each downslope pixel in turn, if a random sample from a uniform distribution spanning 0 to 1 is less than the probability of incoming flow, all flow is directed to that pixel. In general, the D8 flow direction will correspond to the pixel receiving the majority of flow as determined with Tarboton's method, but with sufficient perturbations to allow downslope travel paths to better follow flow directions as indicated by elevation contours. One drawback with this method is that different runs can produce different D8 flow directions for the same pixel. Averaged over large areas, this is of no consequence and the overall accuracy in flow direction is improved, but this method may cause individual debris flow paths to diverge slightly between different runs of the program.

Flow Accumulation

Once flow directions are defined for each pixel, drainage area (flow accumulation) to each pixel can be determined. I use the algorithm described by Tarboton (1997) for flow accumulation, which allows drainage to two lower adjacent pixels. This allows some dispersion of downslope drainage.

It is important to understand that a DEM provides insufficient information to unambiguously define drainage area. The lack of information between DEM points requires that assumptions be made about the shape of the ground surface. Once these assumptions are made, it is possible to define flow lines, but the iterative algorithms employed for estimating flow accumulation do not construct flow lines explicitly. Rather a variety of schemes are used to partition flow to downslope pixels (see e.g., Wilson and Gallant 2000). I am unaware of any assessment of the accuracy of the different schemes, and have chosen one that is both easy to implement and that provides results consistent with expected dispersion and convergence of flow in topographically complex areas (Tarboton 1997).

Once the criteria for channelization (described below) is met, downstream dispersion is no longer allowed and all flow is directed toward one of the downslope pixels. Thus, if need be, the flow direction is altered to align with one of the D8 flow directions (i.e., it must be an integer multiple of PI/4).

I've tried several criteria for determining which of the adjacent pixels to direct *channelized* flow to. The first is the standard algorithm in which flow is directed to the pixel with the path of steepest descent (e.g., Wilson and Gallant 2000). I've found, however, that this strategy can result in misdirected streams in some instances, particularly over areas of relatively uniform surface gradient.

The second criteria is to direct flow to one of the two downslope pixels that would have received flow using the Tarboton algorithm, choosing the one with the greatest topographic convergence, which may or may not coincide with the path of steepest descent. The reasoning for this choice is that stream channels are commonly delineated on contour maps via crenulations in the contour lines, which indicates local topographic convergence. I use the value of *pin* (read from file pin_ID that was created during calculation of flow directions) as an estimate of topographic convergence. This strategy results in better alignment of some DEM-inferred stream channels with contour crenulations, but not always. The primary factor causing misdirected streams in this case is an artifact created in creation of the DEM. The USGS's use of a distance-weighting interpolation algorithm results in high surface curvature values at contour crenulations, but low curvature between contour lines. Where contours are widely spaced, traced stream locations may diverge from the contour crenulations.

A third choice for directing channelized flow is to choose the lowest of the two pixels receiving flow (from the Tarboton algorithm). This works well in some instances, and not so well in others.

None of these strategies is foolproof; all result in misdirected channels. I've had best results using topographic convergence as the criteria for directing flow, but results may vary from one DEM to another.

Drainage Enforcement Using a Channel Mask

Bld_grds can use an optional channel mask to guide channel tracing in areas where the DEM provides insufficient information for determining channel location, e.g., over flat areas. The channel mask is provided as a raster grid, created from a vector line coverage. To guide subsequent tracing of the channel network, elevations within a certain distance of the channel locations specified in the channel mask are lowered. The extent of lowering is a function of distance from the masked channel, and varies linearly from zero for pixels greater than 100 meters from the channel to a maximum at pixels corresponding to the channel mask. The maximum extent of lowering is specified by the value of "dig" in the parameters.dat file.

Lowering of elevations along specified channel pathways is done after the filling of the DEM and prior to determination of flow directions. The sequence followed is 1) create a depressionless DEM, 2) lower elevations along channel pathways, and 3) again create a depressionless DEM, since the lowering of elevations along the specified channel pathways will create new closed depressions.

The degree to which the channel mask is enforced is controlled to some extent by the maximum depth of allowed incision. This depth is specified by the cell value in the channel mask (in integer decimeters). If the value is negative, a constant specified by the value of the parameter "dig" in parameters.dat is used. The larger the allowed incision, the greater is the influence of the channel mask on derived flow directions. However, as the channel mask exerts greater control, locational errors between the line coverage used to create the channel mask and the DEM can result in the placement of channels in inappropriate places – up valley walls, over islands, etc. The topographic attributes derived by Netrace are based on the original DEM elevations (not those altered for removing depressions and enforcing drainage) and are referenced from the channel locations determined by Bld_grds. Hence, misplacement of the channels relative to the DEM will result in incorrect channel gradients, valley-floor widths, valley side slopes, etc.

Mean Annual Precipitation

At the same time that flow accumulation is being calculated, Bld_grds can also accumulate mean annual upstream precipitation. This requires an additional input grid of mean annual precipitation for each DEM point. The resulting mean annual precipitation volume for each pixel is stored in the raster file pval_ID. This file will be used by Netrace to report mean-annual precipitation depth for the drainage area to each reach.

Bld_grds will look for an input grid named "prec_ID", in either IDRISI raster format, ArcView binary raster export format, or as a BIL file. Units are assumed to be in millimeters of precipitation per year. If such a file is not found, no pval_ID file will be produced.

Determination of Channel Initiation Points

There are a variety of published strategies for estimating the location of channel heads using DEMs, e.g., (Tarboton and Ames 2001), (Wilson and Gallant 2000). All employ calculation of some topographic threshold, typically a function of drainage area, that when exceeded indicates presence of a channel. Bld_grds currently employs two criteria, one applied on low-gradient areas where channel expansion occurs primarily through fluvial processes and another applied on high-gradient areas where channel expansion may occur via mass wasting processes. For low-gradient areas we employ a slope-dependent drainage area threshold proposed by Montgomery and Dietrich (1992) and Dietrich et al. (1993);

$$a_{cr} S^{\alpha} = C ,$$

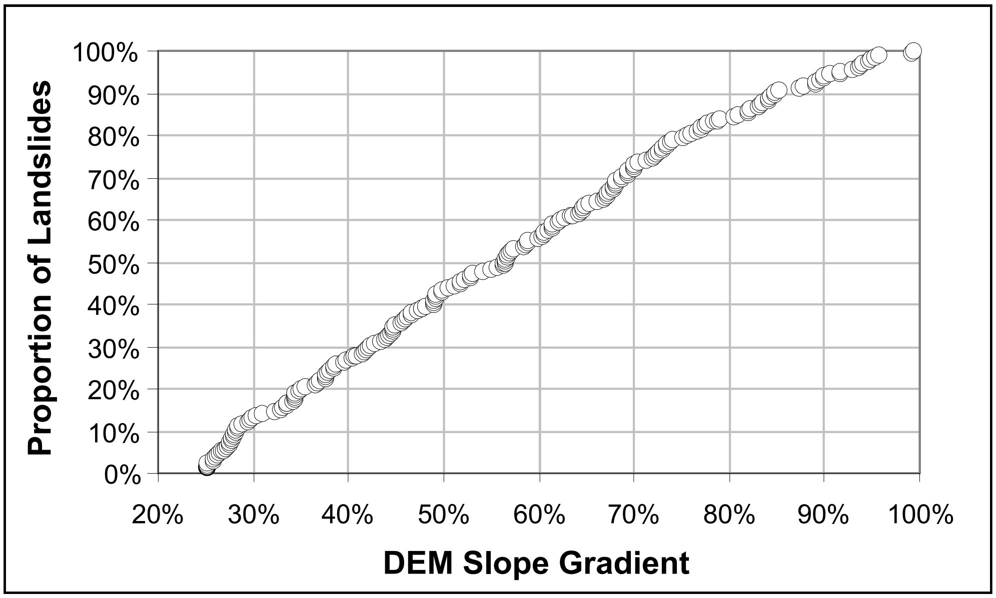

Figure 1. Distribution of surface gradients estimated with a 10-meter DEM at landslide initiation points, Knowles Creek Basin ($60km^2$), Oregon Coast Range.

where a_{cr} is a critical specific drainage area (drainage area per unit contour) required for channel initiation, S is surface gradient, α is an exponent that varies between 1 and 2, and C a constant. Following Montgomery and Foufoula-Georgiou (1993), we use a critical drainage area threshold (rather than a specific drainage area) and set the exponent α to 2. For steeper areas we employ a simple drainage area threshold.

The surface gradient cutoff dictating the choice of methods is specified by the value of S_max in the parameters.dat file. For areas with gradients less than S_max, the slope-dependent criterion is used.

In addition to drainage-area-dependent thresholds for channel initiation we also require a minimum topographic convergence at channel heads (specified by the value of P_min in parameters.dat). This corresponds to enforcing a threshold in contour-line curvature for channel initiation.

Choice of appropriate thresholds (*C* for low-gradient areas, critical drainage area for high-gradient areas, and S_max for separating the two) is confounded by the fact that channel-head locations are dictated by factors other than topography. Moreover, channel heads may migrate up or down stream over time in response to flood and mass-wasting events. The goal, therefore, is to set threshold values that reproduce appropriate channel densities, since accurate reproduction of individual channel-head locations is probably not feasible. I'll discuss strategies for each.

An important point to remember is that the channel density resulting from any choice of threshold parameters is also a function of the horizontal and vertical resolution of the DEM and of its accuracy. Recall also that DEM accuracy reflects the accuracy of the source data. We've found that channel density estimated from 10-meter (horizontal resolution) DEMs with 1-decimeter vertical resolution interpolated from 40-foot contour lines on 1:24,000-scale USGS 7.5-minute quadrangles can vary dramatically from one

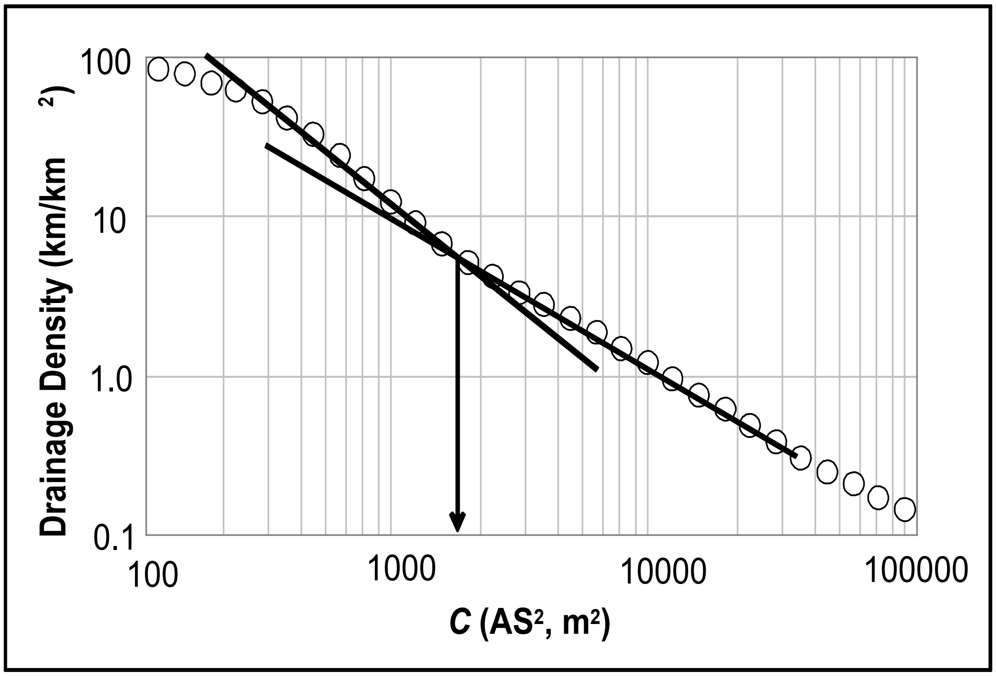

Figure 0 Channel density as a function of *C* for low-gradient areas.
The dots show average for all pixels with Log(AS^2) values falling within bins of size 0.1. The solid lines show local power-function fits to these curves. Their intersection indicates the value for C at which channel feathering begins (i.e., extension of the derived channel network onto unchannelized hillslopes). These results are also for Knowles Creek Basin, Oregon Coast Range.

quadrangle to another depending on the degree to which topographic texture is preserved in the contour crenulations depicted on the maps.

S_max: I view S_max as separating channel initiation into two process domains, fluvial erosion of surface material and mass wasting. Of course, other processes are also active (e.g., seepage erosion) and even these two process types are not cleanly separated by a difference in surface gradient (e.g., gullies form on steep slopes). It is useful to separate channel initiation processes in terms of slope gradient, however, because use of a single criteria across both low- and high-gradient slopes leads to either an under or over-estimate of channel density on one or the other.

To estimate an appropriate value for S_max I use observed mass wasting locations as a guide. The range of slope values over which mass wasting occurs may be found by overlying a point coverage of landslide initiation sites over a grid of slope values estimated from the DEM.

Using the distribution of gradients observed for landslides at Knowles Creek basin, for example, I've chosen a value of 25% for S_max. The minimum end of this distribution is an arbitrary choice; one could just as well argue that catching 85% of the distribution (at a gradient of 35%) is the appropriate level. In any case, use of S_max allows division of the landscape into low- and high-gradient areas over which to apply different criteria for channel initiation.

Low-Gradient Threshold, *C*: Physically, the value of *C* reflects regional properties of soil, bedrock, and climate. This is a complex mix, however, confounded by limits in DEM resolution and accuracy and by heterogeneity in the factors affecting channel formation, so that an *a priori* estimate of *C* is probably not feasible. Hence, we must estimate *C* based on field observations. Montgomery and Foufoula-Georgiou (1993) report a relationship $C \sim 10^6/R_a$ for study areas in California and Oregon, where *C* is in square meters and R_a is mean annual rainfall in millimeters. If you have data on regional channel densities, you can calibrate *C* to your DEMs directly. Another approach, suggested by Montgomery and Foufoula-Georgiou (1993), is to reduce *C* until "feathering" occurs, i.e., extension of the channel network onto unchannelized hillslopes. This method essentially sets *C* to maximize the channel density resolved with the DEM.

A simple means for doing this is to create a slope grid and a flow accumulation grid (with no channelization of flow allowed, i.e., set the *C* value extremely high in parameters.dat) with Bld_grds, from which to create a grid of $A*S^2$ values. Then separate out those pixels with gradient less than S_max, which can be binned and displayed as a cumulative distribution from which to estimate channel density as a function of the threshold *C*.

In Figure 2 I multiplied the number of pixels within a bin by 12 meters (the average channel length within a pixel) to get the total channel length corresponding to the number of pixels in a bin and then divided by the total number of pixels with gradient less than S_max in the DEM, multiplied by 100 meters (the area of a pixel). This provided an estimate of channel density. As expected, channel density increases as the value of *C* decreases. At a *C* value of about 1500 square meters the rate at which channel density changes with decreasing *C* increases, shown by the change in slope on the log-log plot. This is where inferred channels start to extend up portions of the hillslope with no contour crenulations – channel "feathering" – leading to a dramatic increase in channel density. I thus chose 1500 square meters as an appropriate value of *C* for this DEM, which gives a channel density of about 6 km/km^2, a value in line with observed channel densities in the Oregon Coast Range.

High-Gradient Areas: The intent here is to find a drainage-area threshold appropriate for mass wasting processes. A more process-based strategy could be employed, something that employed slope as well, but I expect that mass-wasting controls on channel initiation vary dramatically (with each storm season), so I've opted for the simplest criterion.

Again, the goal is to set a threshold value that reproduces an appropriate channel density, not to accurately replicate specific channel-head locations. For that purpose, the same strategy described above for low-gradient areas may be used. Create a flow accumulation grid with Bld_grds (with the channel_area_threshold set to an extremely high value in parameters.dat, to preclude flow channelization); bin pixels with gradients greater than or equal to S_max, and plot a cumulative distribution in terms of number of pixels verses accumulation area.

Surface Gradient

Surface gradient for each DEM pixel is calculated using a finite difference scheme as described in Wilson and Gallant (2000). The equation for slope is equivalent to that

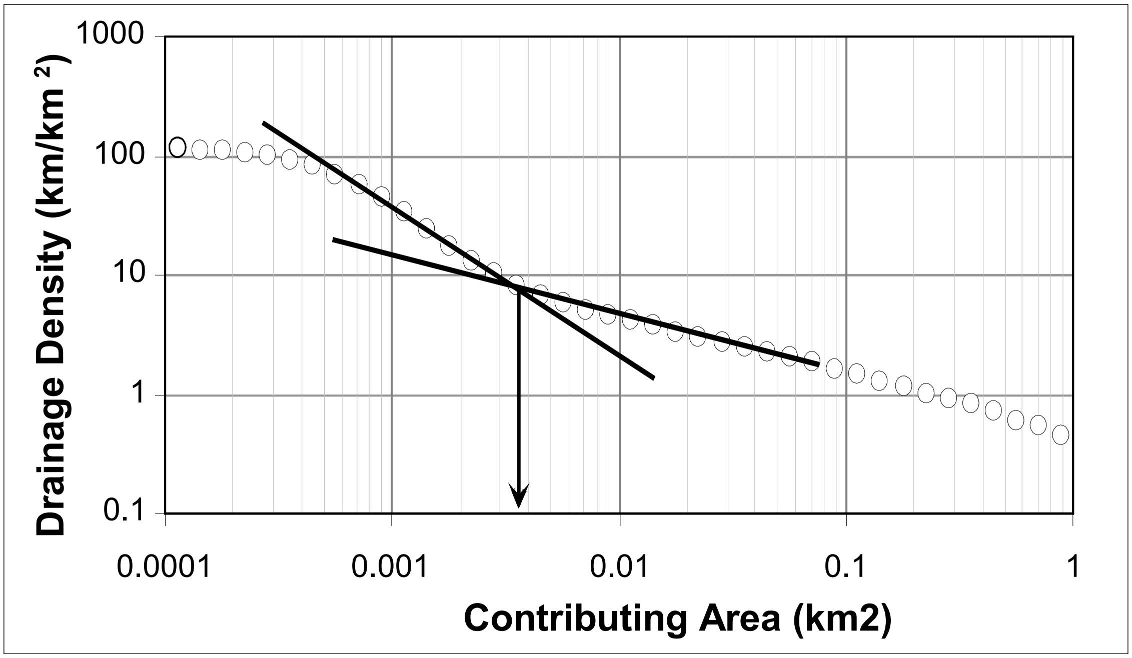

Figure 3 Channel density as a function of contributing-area threshold for high-gradient areas. Dots indicate mean value for all pixels with Log(A) falling within bins of size 0.1. The inflection point at about 0.04 km^2 indicates the area threshold below which channel feathering occurs.

described by Zevenbergen and Thorn (1987) and uses adjacent pixels in the four cardinal directions.

Dietrich et al. (2001) point out that this method for estimating surface gradient produces an error that varies with slope azimuth and therefore advocate another method that utilizes more of the adjacent pixels. In testing these algorithms against a synthetic DEM, with known surface gradients at every point, I've found that there is indeed a directional anisotropy in the error associated with the finite difference calculation of slope. The anisotropy is reduced by using all eight adjacent pixels, however, the absolute error is increased. I have chosen to use the method that exhibits directional variance, but has greater accuracy.

Specific Drainage Area: Contour Length

Specific drainage area (drainage area per unit length of contour) is used in some algorithms for estimating water flux (both overland and sub-surface). Drainage area for each pixel is estimated as described above, the contour length for each pixel is estimated as described here and stored in raster file bcont_ID in units of number of pixel lengths. Contour length is estimated during the calculation of flow directions using the Tarboton algorithm (Tarboton 1997). A flow direction is calculated for each of the eight triangular facets defined by the current DEM pixel and its eight neighbors. For facets with flow exiting the pixel, contour length is estimated as the projection of the flow on the edge of the facet. Thus flow at an angle of θ to the edge spans a contour of length ½ $\mathrm{Sin}(\theta)$ pixel lengths; flow perpendicular to the edge crosses a contour of ½ pixel length. The equivalent contour length for each facet with flow exiting the pixel is summed, and this value is written to bcont_ID. This provides an estimate of the total contour length crossed by flow exiting the pixel. The total drainage area for the pixel divided by this value provides an estimate of specific drainage area.

NETRACE

Netrace uses the raster files created by Bld_grds to trace a channel network, which if requested, is extended to estimated shallow landslide initiation sites. The DEM is then used to estimate a variety of topographic attributes for each pixel identified as a channel. These are written to an ArcView shape file. The methods used for calculating these attributes are described below.

Command-Line Prompts

Netrace requires specification of a "Root Directory" (path to the data files) and then asks for an "Input ID:" and an "Output ID:". Shape files are created with the output ID, so that by using different input and output Ids, separate shape files can be made from the same set of input files. You are then offered four options for delineation of the watershed:

1) Delineate the maximum drainage area
2) Specify the drainage area to delineate
3) Specify a single pour point
4) Iterate over DEM to get all channels

Since a single DEM may contain any number of separate watersheds, these options provide ability to chose which one(s) to trace. Options 1 and 2 require no further user input; option 2 requires input of the drainage area (in square kilometers) to delineate; option 3 requires input of the DEM column and row numbers of the pour point (the mouth of the basin you want to analyze). You'll be prompted for any additional information required.

Trace then performs the set of operations specified in the instructions.txt file. Screen output reports on program progress.

Outputs

The channel network is subdivided into a series of reaches. Reach endpoints are placed at channel tributary junctions. Where tributaries are spaced far apart, reach endpoints are positioned to provide reaches of relatively uniform length and uniform attributes (e.g., channel slope, valley width). Channel locations are provided in an ArcView (line) shape file and reach attributes are provided in a dBase file. The attributes that can be estimated for each reach are listed below. A (point) shape file that provides information about tributary junctions (e.g., junction angle) can also be created. The tributary-junction shape file provides attributes that may correlate with the size of debris fans created at tributary mouths.

An additional comma-delimited ASCII file is also created, named routes_ID.txt, which lists the LLID, beginning channel length (which is always zero), and the final channel length for each channel. This file expedites creation of a routed channel coverage from the shape file in Arc.

Netrace, as discussed further in the following section, also includes a set of subroutines for estimating debris-flow inundation areas. Results from these routines are provided as a raster file that identifies landslide source areas, zones susceptible to debris-flow traverse, and areas susceptible to inundation by debris flow deposits.

Shape File Attributes for Reaches

Output shape files are named "reach_ID" and include the following attributes.

Length: Reach length in meters. Note that the reach length specified here may not match exactly the reach length measured from node to node in the vector file. The vector-file vertices are placed at pixel centerpoints, where as Netrace estimates channel length using channel locations that may not cross the DEM pixel midpoint, in an attempt to provide a more accurate estimate of total channel length.

Area: Drainage area in square kilometers to the downstream end of the reach.

LLID: Each channel is assigned a unique identifier, based on the latitude and longitude at the mouth. At channel junctions, the identifier is assigned to the channel with the largest drainage area and the smaller channel is assigned a new LLID. LLIDs are calculated using pixel corners to differentiate cases where two tributaries enter the mainstem at a single DEM node.

From_Dist: The distance from the channel mouth to the downstream end of the reach (in meters).

To_Dist: The distance from the channel mouth to the upstream end of the reach (in meters).

Order: Strahler stream order. Note that stream order is a function of where the channel head is defined.

Mean_Grad: Mean gradient through the reach.

Max_Grad_D: Maximum gradient encountered downstream through the network.

ValWidth_L: Valley width in meters on the left side of the channel (facing downstream).

ValWidth_R: Valley width in meters on the right side of the channel.

ValSlp1_L: Estimated surface gradient at the base of the valley wall on the left side of the channel (facing downstream), over a horizontal length of ~10 meters.

ValSlp1_R: Estimated surface gradient at the base of the valley wall on the right side of the channel, over a horizontal length of ~10 meters.

ValSlp2_L: Estimated surface gradient of the valley wall on the left side of the channel over a horizontal length of ~100 meters.

ValSlp2_R: Estimated surface gradient of the valley wall on the right side of the channel over a horizontal length of ~100 meters.

Lake: ID of any lake (specified in the raster mask lake_ID) crossed by the reach; zero if there is no lake.

MnAnPrc_mm: Mean annual precipitation depth in millimeters, averaged over the drainage area to the downstream end of the reach.

Dep_Prob: Estimated probability of debris flow deposition in the reach.

Scr_Prob: Estimated probability of debris flow scour through the reach.

Dep_Sum: Volume estimate of debris-flow-deposited material found in the reach. Relative units: large values indicate high probability of finding a large volume of debris flow material.

Shape File Attributes for Tributary Junctions

MstmAr(km2): Drainage area of the mainstem, in square kilometers (assumes 10-meter DEM), at the tributary junction.

MstmGrad: Channel gradient of the mainsteam at the tributary junction.

MstmValWth: Estimated vallel width of the mainstem at the tributary junction.

JnAngl(deg): Junction angle of the tributary with the mainstem, in degrees

Expctd_LS: The expected number of landslides to be found in the tributary basin, based on integrating landslide density (summing pixel by pixel) over the drainage area of the tributary

LS_delivry: A relative volume estimate for debris flows at the tributary mouth, same as *Dep_Sum* above, but at the tributary mouth.

Trib_Area: Drainage area of the tributary basin in square kilometers.

Algorithms

LLID

A unique 13-digit identifier, derived from the digital latitude and longitude (to a precision of 1-ten-thousanth of a degree) of the mouth of the channel, is assigned to each reach. The identity of the "main stem" at tributary junctions is based on drainage area: a new LLID is assigned to the channel with the smaller of the two drainage areas. Use of the LLID allows grouping of all reaches belonging to a single channel and is required by Arc for creation of a routed channel network.

Since most DEMs are provided in UTM coordinates, the projection to latitude and longitude is different for the two horizontal datums commonly used. The correct datum for the DEM (NAD27 or NAD83), along with the UTM zone number of the DEM, is specified in the parameters.dat file.

Channel Length

Channel orientations are estimated point-by-point by fitting of a quadratic through a window centered over the channel (DEM) point. The length of the window is twice the "junction length" specified in parameters.dat. Hence, a 50-meter junction length will result in fitting of a 2nd-order curve through 100-meters of channel, which will include 7 to 10 points (in a 10-meter DEM), depending on the channel orientation. The intersection of this curve through the center pixel is then used as an estimate of channel length through the pixel. This gives on average shorter lengths than obtained by assuming that channels cross directly through the center of the pixel. This procedure also provides an estimate of channel orientation (flow direction) through each pixel.

Channel lengths show up in three fields of the dBase file that is produced: the length of the reach (in meters) and the “from” distance and “to” distance, which specify the distance from the mouth of the channel for the downstream and upstream-ends of the reach, respectively. The “from” and “to” distances are used by Arc to create a routed coverage. Note that these lengths will vary from those calculated by ARC from locations of nodes and vertices, since they align with the DEM points, whereas channel lengths have been estimated as described above.

Channel Gradient

Channel gradients are estimated using a polynomial fit to channel pixel elevations over a centered window. Window length is varied as a function of channel slope and channel points falling below the fitting curve are weighted preferentially (assuming that these DEM points fall closer to the actual channel). This procedure is iterated until successive estimates of gradient converge (to some small tolerance) at all points. The resulting gradients closely match those found using contour lines on the 1:24000-scale base maps (from which the DEMS were derived).

The distances over which to apply the centered window are specified by the values of Xmin and Xmax in parameters.dat. These distances are applied at channel gradients specified by Smin and Smax; the length of the window varies linearly as a function of channel gradient between the values specified by Smin and Smax and are held constant for gradients greater or less than these limits. The order of the polynomial used for the fit is also specified in parameters.dat.

It is important to ensure that the length of the window is sufficient to include a number of DEM pixels greater than the number of points required for the polynomial fit. We want to *over* fit the polynomial. The longer the window specified, the greater will be the number of included points, and the greater will be the amount of smoothing applied to the fit channel profile. It is also appropriate to use polynomials of relatively low order, e.g., no greater than 3. Higher-order polynomials tend to result in spurious high-gradient reaches.

An estimated channel gradient is obtained for each channel pixel. These are then averaged over the length of the reach and reported in the field “Mean_Grad” in the dBase file. The largest gradient encountered downstream (to the edge of the DEM) is also recorded, in the field “Max_Grad_D”. This value may be useful for culling reaches that lack fish presence because of a downstream gradient barrier.

A problem encountered with calculation of channel gradient is creation of short, steep reaches that result from errors in the DEM. When channel gradient is used to indicate fish usage, such reaches may incorrectly indicate barriers to fish passage. One way to potentially overcome this problem is to force greater smoothing of the fitting polynomial by increasing the values of Xmin and Xmax and by using a very low-order fit, e.g., order 1.

Valley-Floor Width

Width of the valley floor is estimated as the length of a transect that intersects the valley walls at a specified height above the channel. Since the orientation of the valley is unknown, transect orientation is varied to find that which provides the minimum length. The height above the channel is specified as a 2.5 times estimated bank-full depth, given as a function of drainage area. This value is currently coded in (cannot be altered) as

$$H_{bf} = 0.36A^{0.2}$$

where H_{bf} is bank-full depth and A is drainage area in square kilometers (based on data in (Benda 1994)). Thus, the height above the channel from which the transect is extended varies from a little less than a meter for small streams to a little over three meters for large streams. Elevations are linearly interpolated between DEM points. The accuracy of these estimated widths is directly dependent on the resolution and accuracy of the DEM.

An estimated valley-floor width is obtained for every channel pixel, one for each side of the channel. Since there are occasionally some pixels where this strategy fails – the transect may be incorrectly oriented at tributary junctions, for example – I then check for outliers in the estimated width over a centered window that spans 10 pixels. Any values exceeding 2.5 times the median are considered in error and are replaced by a linear fit through the remaining points. The resulting widths are then averaged over the length of the reach.

Valley-Wall Gradients

Two estimates of surface gradient are made for each side of the valley wall. One estimate applies to the base of the wall and is calculated from the change in elevation over a 10-meter distance from the intersection of the transect used for estimating valley width. This value roughly approximates the steepness of the valley wall adjacent to the valley floor. The other estimate is based on the change in elevation over a 100-meter distance from the base of the wall. This provides a mean slope over a longer length and may be useful as a predictor of channel-adjacent mass-wasting potential. To determine the orientation of the valley wall (so that the elevation change is taken parallel to the aspect of the slope), I sample two points along a transect oriented normal to the estimated slope azimuth, each 50 meters from the midpoint of the 100-meter line extended up from the base of the valley wall. Orientation of the azimuth is varied to minimize the difference in elevation of these two points. This provides an estimate of valley-wall aspect over a length scale of 100 meters.

These two estimates of surface gradient are made for each side of the valley for each channel pixel. Again, there are opportunities for mistakes here, particularly at tributary junctions. Hence, the procedure for culling outliers is repeated with each of the slope measurements. The values reported for each reach then represent the average of the remaining points.

Tributary Junction Angles

The angle with which tributary channels enter the mainstem is estimated from the channel orientations found when fitting quadratics through windowed lengths of channel for estimating channel length (see the section, “Channel Length”, above). Through the mainstem, this window is centered about the junction pixel. Through the tributary the window extends upstream from the junction.

Debris-Flow Potential

Subroutines that implement an empirical model for estimating probability of debris flow impacts to stream channels is included in Netrace. Two quantities must be derived to estimate potential for debris flow impacts: 1) the upslope susceptibility to debris-flow-triggering landslides and 2) the probability that a debris flow initiated up slope will travel to the channel reach.

Within Netrace, landslide susceptibility is treated in terms of the probability of encountering a landslide scar within a pixel. This probability may be derived from an estimate of landslide density (number of landslides per unit area). A high density implies a high susceptibility to landsliding. For the current implementation, landslide density is empirically calibrated as a function of a topographic index and vegetation cover. However, any method can be used to estimate landslide density. Netrace will look for a file named denls_ID.rst (or denls_ID.flt) in the data directory. If it finds it, it will read the corresponding densities (in terms of the number of landslides per pixel) from this file. If it doesn't find it, it will calculate densities using the default relationship between a topographic index of slope instability based on SHALSTAB, stored in a file named qt_ID. The default relationship also requires a file specifying vegetation type for each pixel, in terms of a classification used by the Coastal Landscape and Modeling Study (CLAMS). A description of this relationship will be presented in another document.

The probability that a debris flow will travel from any specific landslide source pixel to a downslope channel reach is estimated from an empirical model currently calibrated to the Oregon Department of Forestry data for debris flow impacts from the 1996 storm (Robison, Mills et al. 1999). This model, derived from the Benda-Cundy model (Benda and Cundy 1990), specifies a probability of runout from any specific landslide initiation site as a function of three topographic parameters and as a function of riparian stand type. The topographic parameters are 1) channel gradient, 2) cumulative upstream scour length, and 3) junction angles at tributary junctions. These are derived from the input DEM. Additionally, the probability that the debris flow erodes material from the bed and the probability that it deposits material are also estimated. Riparian stand type is based on the CLAMS vegetation coverage. This model is described in Miller et al., (in prep). The current implementation has no adjustable parameters. Future updates will include programs for calibrating the model and ability to adjust model parameters within Netrace.

Once probabilities of landslide initiation and debris flow runout are made for every potential landslide initiation site, these can be routed to the channel and summed to provide an estimate of the probability of debris flow scour of the reach, stored in the field "Scr_Prob" of the dBase file, and the probability of debris flow deposition in the reach, stored in field "Dep_Prob". A relative estimate of the volume of debris-flow-derived sediment to be found in the reach is provided by the value of "Dep_Sum" in the dBase file. Dep_Sum is calculated as the cumulative scour length of all debris flow tracks leading to the reach times the probability of initiation and the probability of runout to and deposition in the reach. Its units are length, in terms of the cumulative number of pixels of scour length, but should be interpreted as relative volume, e.g., large values indicate a greater probability of finding a large volume of debris-flow-transported material in a reach.

Debris Flow Inundation Hazards

With the help of Jon Hofmeister (Oregon Department of Geology and Mineral Industries), a model for estimating a relative potential for debris flow inundation is implemented in Netrace. The model is based on that of Iverson et al. (1998), which was originally presented for estimating lahar inundation hazards. Iverson et al. showed that lahar cross section and inundation area can be estimated with the equations

$$A_{cs} = aV^{2/3},$$
$$B = bV^{2/3};$$

where A_{cs} is cross-sectional flow area of the lahar, B is the area inundated by the lahar, V is total lahar volume, and a and b are empirical coefficients. Iverson et al. suggest that these same equations may be applied for colluvial landslide-triggered debris flows, with proper calibration of the two coefficients. Schilling et al. (1998) describe an implementation of the model in Arc. Building on the work of Iverson et al. and Schilling et al., we've implemented this model for estimating debris flow inundation hazards using a DEM.

To address debris flow hazards requires examination of a multitude of potential landslide initiation sites and runout tracks and estimation of deposit volume for each. Initiation sites are empirically identified as functions of a topographic index. Two choices for this index are currently provided: surface gradient and output from SHALSTAB. Currently the index is used simply to provide a threshold value for identifying pixels susceptible to landsliding, i.e., using slope, if surface gradient exceeds the threshold, the pixel is considered susceptible to landsliding; for SHALSTAB, landslide susceptibility is assumed if the steady-state precipitation value for failure is less than a specified threshold. The threshold value is specified by the parameter Istar_max in parameters.dat. The choice of index to use is specified in the instructions.txt file.

Lacking data other than elevations, we've chosen to estimate potential deposit volume based on the cumulative scour length of each track, multiplied by the estimated mean volume of colluvium available for scour per unit length (specified in parameters.dat). Using the model presented by Benda and Cundy (1990), potential debris flow tracks are divided into zones of scour, transport (with no net scour or deposition), and deposition. Parameter values for this model are specified in parameters.dat. Every debris flow track thus has an estimate of the maximum deposit volume that might traverse it, based on the cumulative upstream scour length.

Once a depositional zone is reached, valley cross-sectional area is estimated as a function of inundation depth (using the DEM). This is then translated to an estimate of debris flow volume as a function of inundation depth using the equation above. On debris flow fans and on large valley floors, cross-sectional area is estimated as a function of deposit width by assuming a constant surface slope normal to the flow direction. Every DEM pixel within vicinity of a potential debris flow track thus has an estimated minimum debris flow volume required for inundation.

Then, moving downslope along each debris flow track, we accumulate a sum of all pixels as a function of volume, i.e., we create a cumulative distribution of the volume of pixels potentially inundated so far. If the area inundated exceeds that corresponding to the volume assigned to a particular pixel (*B* in the equation above), the pixel is assigned the next largest volume for which the area inundated is still less than that corresponding to the maximum volume available. Each pixel is then assigned a value equal to its assigned or reset volume divided by the maximum potential volume of a debris flow deposit reaching that point. This ratio serves as an index of relative inundation hazard; the smaller its value, the greater the likelihood of that location being impacted by deposition from a debris flow. This procedure is repeated for all potential debris flow tracks and all channels. Where deposit zones overlap, the smallest value (highest hazard) is maintained.

Output is to an ASCII raster file named hazard_ID.grd. Values are coded differently depending on the choice of topographic index for instability. Codes are as follows.

Slope

0-99: Surface gradient in percent. Higher values imply greater landslide susceptibility. To translate to hazard ratings requires an empirical calibration, as described above.

100: Debris flow transport (with a 1-pixel buffer).

101-200: Relative inundation ratio times 100 plus 100. Lower values imply greater hazard. Interpretation requires field calibration, but for example: 101-115 – high hazard; 116-150 – moderate hazard; 151-200 – low hazard.

SHALSTAB

0-100: Ratio of minimum inundation volume to maximum potential volume times 100. This provides a measure of debris flow inundation hazard – the lower the value, the higher the hazard. Interpretation requires field calibration, but an example would be: 0-15 high hazard, 16-50 moderate hazard, 50-100 low hazard.

107: Debris flow transport zone, with a 1-pixel buffer.

116-9999: Istar (topographic index) value, lower values correspond to higher landslide susceptibility. Interpretation requires field calibration to a landslide inventory.

INPUT PARAMETER FILES

Parameters.dat

Adjustable parameters for Bld_grds and Netrace are held in the ASCII file parameters.dat. A description of each parameter is provided below.

Sl the number of pixels over which slope is calculated (> 1 to address "pocket terracing"). The weighted-distance algorithm used by the USGS and many contractors to interpolate line-trace 10-meter DEM elevations between contour lines results in spurious variations in the first and higher-order derivatives of elevation. Interpolated elevations tend toward flatter topography in the vicinity of a contour line (where data density is greater) with much of the variation occurring between contours. When estimating derivatives such as slope or topographic convergence, this artifact results in bands of high slope, or high convergence, parallel to the contours. These bands can be seen in shaded relief images produced from these DEMs and are particularly apparent as a set of apparent pits, or “pocket terraces” along lower-gradient channel courses. These artifacts are troublesome for algorithms that use estimates of slope and topographic convergence, such as estimates of relative soil saturation and slope stability. To address this issue, I’ve added the option for some smoothing of the derived slope estimate. The value of sl specifies the number of pixels to use as the length increment for calculating surface gradient. I’ve typically used values of 2 or 3. Too many, and you loose resolution; too few and you’re plagued with bands of anomalously high slope gradient. Use as low a number as you can.

channel_area_threshold the contributing-area threshold for channel initiation in high-gradient areas (areas with surface gradient greater than or equal to *S_max*). Methods for estimating this value using the DEM are discussed in the section above “Determination of Channel Initiation Points”. Given in the same units as used for the DEM, e.g., square meters. Typical values for western Washington and Oregon fall within the range of a few hectares or less (for 10-meter DEMs). Lower values result in higher estimated channel density.

C_min Used to determine the contributing-area threshold as a function of surface gradient for low-gradient areas (those with surface gradient less than *S_max*): $A = C\_min/S^2$, where A is threshold contributing area and S is surface gradient. Typical values for western Washington and Oregon fall within the range of a few thousand square meters or less (for 10-meter DEMs). Lower values result in higher estimated channel density.

S_max The minimum surface gradient for landslide potential, calibrated to a DEM with a landslide inventory. This value is used to separate low-gradient areas, for which a slope-dependent drainage threshold (*C_min*) is used for channel initiation, from high-gradient areas, for which a drainage area threshold (*channel_area_threshold*) alone is used for channel initiation.

P_min The minimum number of inflowing cells required for channel initiation. This value enforces a topographic convergence threshold, in addition to the drainage area threshold discussed above, for channel initiation. Lower values resulting in higher estimated channel density.

debris_flow_slope Used in the Benda-Cundy (Benda and Cundy 1990) model for estimating debris flow runout. This value specifies the channel gradient above which channelized flows don't stop at tributary junctions. Typically set to TAN20° (0.364).

erode_slope Used in the Benda-Cundy model for debris flow runout. Specifies the channel gradient above which debris flows erode the channel bed. Typically set to TAN10° (0.176)

thetamin Used for the Benda-Cundy model for debris flow runout. *Thetamin* specifies the channel gradient at and below which debris flows deposit material. Unlike the Benda-Cundy model, the gradient for debris flow deposition is made a function of estimated valley-floor width. This allows a steeper slope to be specified for unchannelized areas, such as where debris flow channels enter steep, unchannelized debris fans. *Thetamin* is typically set to a value of TAN3.5°(0.06).

wmin Used for the Benda-Cundy model for debris flow runout. The valley-floor width at and below which *thetamin* applies. Given in DEM units (e.g., meters). Typically set to a value of 30 meters.

thetamax Used for the Benda-Cundy model for debris flow runout. *Thetamax* specifies the channel gradient at and below which debris flows deposit material where valley-floor width is greater than *wmax* (below). Typically set to a value of 0.15.

wmax Used for the Benda-Cundy model for debris flow runout. Specifies the valley width at and above which *thetamax* applies. For valley widths between *wmin* and *wmax*, the channel gradient for debris flow deposition is varied linearly between *thetamin* and *thetamax*.

lstop_max Specifies the maximum length for unchannelized, low-gradient (less than *debris_flow_slope*, above) debris flow runout. If the distance from a source pixel to a channel exceeds *lstop_max*, the source pixel is assumed incapable of producing a debris flow that will travel to a channel, i.e., any landslides stop prior to encountering a channel. A typical value for *lstop_max* is 50 meters, for a 10-meter DEM (*lstop_max* must be correspondingly longer for DEMs of lower resolution). Note that for estimating debris flow inundation areas, *lstop_max* should be set very high, 5000 meters say, to ensure that all potential landslide runout tracks are included, even those that do not enter a channel.

colluvial_volume Used for estimating debris flow inundation areas. Specifies the average colluvial volume available for debris-flow scour per unit length of low-order channel. Given in DEM units, e.g., m^3/m. A typical value is in the range of 6 to 8 m^3/m.

colluvial_volume_0 Used for estimating debris flow inundation areas. Specifies the average volume available for debris flow scour per unit length of traverse over unchannelized hillslopes.

Xmin Specifies the minimum window length used for channel gradient estimation, in DEM units. Typically on the order of 100 meters for a 10-meter DEM.

Xmax Specifies the maximum window length used for estimating channel gradient. Typically on the order of 300 meters for a 10-meter DEM

Smin Specifies the channel gradient at and below which *Xmax* applies. Typically set to 0.01 or lower.

Smax Specifies the channel gradient at and above which *Xmin* applies. Typically set to 0.20. Window length varies linearly from *Xmin* to *Xmax,* as a function of channel gradient*,* for gradients between *Smax* and *Smin*.

Fit_Order The (integer) order of the polynomial used for fitting channel profiles. Values of 1 to 3 work well.

junction_length The channel length used to estimate channel flow direction and tributary junction angles, in DEM units. Typical value: 50 meters.

junction_angle Used with the Benda-Cundy model for debris flow runout. Specifies the tributary junction angle above which debris flows deposit (unless the receiving channel gradient exceeds *debris_flow_slope*). Typical value: Tan70° (2.745)

trigger_angle Used with the Benda-Cundy model for debris flow runout. Specifies the junction angle for 1st-order tributary channels above which debris flows deposit. Typcial value: TAN45° (1.0).

Istar_max Threshold value of the topographic index for slope instability. Pixels with index values greater than (or less than, depending on which topographic index is used) are considered source areas for landsliding.

width_coefficient_1 Used for estimating channel width: $W = w_1 + w_2A^2$, where w_1 is *width_coefficient_1* and w_2 is *width_coefficient_2*. This equation is based on a regression to regional data in coastal Oregon (Sharon Clark), for which w1 = 1.6066. Channel widths are only used as a minimum for valley-floor width estimates.

width_coefficient_2 Sharon's regression gives $w_2 = 2.05183$

diffusional_coef_hill An estimate of soil creep rates: $q_s = KS$, where q_s is soil flux, S is surface gradient, and K is the *diffusional_coef*. Separate values of K are allowed for soil flux on hillslopes (e.g., into topographic hollows, using *diffusional_coef_hill*) and for soil flux into channels. Given in DEM units, e.g., cubic meters per meter per year. This value is used for estimating sediment yield.

diffusional_coef_channel Separate creep rates allowed for hillslopes and channel banks; *diffusional_coef_channel* specifies the rate for channels.

coarse_fraction The fraction of colluvium with grain size > 0.25mm. Used for estimating mean annual suspended- and bed-load values.

abrasion_rate Proportion of volume lost per kilometer of fluvial transport. Used for estimating mean annual suspended- and bed-load values.

Minimum_reach_length Reach endpoints for shape-file output are determined first by channel junctions. If the distance between tributaries is sufficiently long, additional reaches will be created. For these, endpoints are positioned so as to create relatively homogenous reaches, e.g., with relatively uniform channel gradient and valley width. Between tributary junctions, reach lengths will try to be maintained within the guidelines specified by the *minimum* and *maximum_reach_lenth* parameters. This preferred reach length is allowed to vary as a function of drainage area. Given in DEM units (e.g., meters).

maximum reach length As above, in DEM units.

Amin The area in square kilometers at and below which *minimum_reach_length* is used.

Amax The area in square kilometers at and above which *maximum_reach_length* is used. The preferred reach length varies linearly from *minimum_reach_length* to *maximum_reach_length* for drainage areas between *Amin* and *Amax*. Common values may be *minimum_reach_length* = 100 (m), *Amin* = 1 (km^2); *maximum_reach_length* = 300 (m); *Amax* = 100 (km^2).

<u>SHALSTAB</u>: Parameters.dat also specifies values for use with the SHALSTAB algorithm when it is used as the topographic index for slope instability. These are used only to translate the topographic index to units of mm/day steady-state rainfall so that output can be compared directly to other implementations of SHALSTAB.

Soil transmissivity (square meters per day)

Soil Saturated Bulk Density (kilograms per cubic meter)

Soil Friction Angle (degrees)

Soil Cohesion (Pascals)

<u>UTM grid information</u>. These values are needed for calculating the LLID – the latitude and longitude of the channel mouths – for each reach.

Datum 83 for NAD83, 27 for NAD27

UTM zone number e.g., 10

<u>Drainage Enforcement</u>. When a channel mask is provided with negative values used to flag channel locations, the "*dig*" parameter is used to specify the maximum depth to which DEM elevations may be reduced for drainage enforcement. Larger "*dig*" values result in greater enforcement to the channel mask, but may also result in spurious channel elevations where errors in the vector channel coverage used to create the mask direct channels onto hillslopes. Values around 1 meters.

Runout_parameters.txt

a_coef_0: The coefficient for cross-sectional flow area. Different coefficients may be used for channelized and unchannelized debris flows; a_coef_0 is used for debris flow tracks traversing hillslope pixels that are not classified as channels. Values less than 0.5 are found to work acceptably well, but field verification is required.

b_coef_0: The coefficient for debris flow inundation area. Here too different coefficients may be used for channelized and unchannelized debris flows: b_coef_0 is used for debris flows traversing hillslope pixels that are not classified as channels. Values less than 100 work acceptably well.

a_coef_c: The coefficient for cross-sectional flow area for debris flow tracks traversing pixels classified as channels.

b_coef_c: The coefficient for debris flow inundation area for pixels classified as channels.

Radius: Used in creating transects for calculating valley cross-sectional area. First, every hillslope pixel is associated to the nearest channel pixel. Then a circle with a radius of the specified number of pixels is constructed about every channel point. Pixels intersected by the edge of this circle serve as endpoints for the transects. This is done both for a circle of the specified radius and for a circle with radius given by *radius*/5. This procedure produces multiple potential valley-crossing transects for every channel point. The one producing the smallest cross-sectional area is used. With 10-meter DEMs, I've found a *radius* of 25 pixels works well.

Max_bin: The largest ratio of inundation volume to maximum potential debris flow volume to continue estimating inundation hazards for. The larger this value, the greater will be the number of pixels for which the ratio is calculated, however, a large number of them will be for a very low hazard potential. I've found a max_bin value of 1.0 works well.

Nbins: The number of bins to use for tracking the cumulative distribution of debris flow volumes associated with pixels inundated as we follow a debris flow track downstream. A larger number of bins provides greater resolution of hazard variation, but requires greater processing time. I use a value of 256, since this is (for most plotters) the maximum number of color variations that can be plotted.

de_min_0: Used for estimating deposit cross sectional area as a function of inundation width over flat areas. Cross sectional area is estimated as

$$A_{cs} = de\_min \cdot W^{rp}$$

where W is deposit width, measured from the channel, and rp is another adjustable parameter, usually set to 1.0 (in which case, cross sectional area corresponds to a deposit with a constant surface slope equal to *de_min*). Separate values for *de_min* may be specified for channelized and nonchannelized debris flows; *de_min_0* applies to hillslope pixels not classified as channels. Values of *de_min* are typically in the range of 0.05 to 0.15.

de_min_c: The value of *de_min* to use for pixels classified as channels.

de_max: Valley transects are terminated if elevation along the transect decreases by an amount greater than de_max, in which case it is assumed we've crossed a drainage divide. I typically set de_max to 5 meters.

rp: The exponent on W in the equation above for estimating cross sectional area through flat areas. A value of 1 corresponds to a deposit with a constant surface slope equal to *de_min.*

smooth_output: (y/n) Specifies whether the inundation ratio is smoothed prior to output. Smoothing is done over a three-by-three pixel weighted average excluding pixels with zero or nodata values. Weighting is based on (1-B), where B is the inundation ratio of the pixel. Hence, high-hazard pixels are preferentially weighted.

Buffer: The size of the buffer to add around inundation hazard areas, in pixels. Buffer pixels are added based on the average inundation-ratio value of adjacent pixels with nonzero values.

The Instruction File

The file instructions.txt tells program Netrace which attributes to calculate and write to output files. Some of the lines in here are used for things not described in this document, like creation of input files for ESI's simulation programs. I'll describe what each of the lines implies below. Note that when a shape file for reaches is created, certain attributes are automatically written to the output dBase file, while others are optional. Those that are automatically written are: reach ID, LLID, reach length, drainage area to the downstream end of the reach, to distance, and from distance.

```
Instruction file for Netrace
===================================
 run   subroutine
(y/n)  (name)             task
-------------------------------------
y      out_order          create image grid with stream order
```

A "y" here will have stream order included in the output shape (or IDRISI) file.

```
y      aspect             estimate channel flow direction
y      node_length        estimated channel length in each node
```

These lines are a legacy of an older version; these attributes are automatically calculated.

```
y      channel_gradient   Calculate gradients
```

A "y" here results in calculation of channel gradients. This is somewhat time consuming, so if gradients are not needed, you have the option of not taking the time to calculate them.

```
n      out_gradient       Create output image
```

A "y" here results in creation of the two gradient fields (Mean_Grad and Max_Grad_D) in the output shape file (or the output IDRISI file)

```
y      debris_flow_routing_v Debris flow delivery
```

A "y" here results in calculation of debris-flow-impact probabilities.

```
y      out_debris_flow    Create output image
```

A "y" here results in creation of the three debris-flow-related fields (Scr_Prob, Dep_Prob, Dep_Sum) in the output shape file.

```
y      valley_form
```

A "y" here results in estimation of valley width.

```
n      out_valley_form
```

A "y" here results in creation of the two valley-width fields (ValWidth_R and ValWidth_L) in the output shape file.

```
n      creep
```

A "y" here results in calculation of soil-creep inputs to the channel network. No output options are provided; these estimates are used subsequently for estimates of yield and for creation of simulation input files.

```
n      yield
```

A "y" here results in calculation of mean annual sediment yield estimates for total load, suspended load, and bed load. Not implemented in all versions.

```
n      ARCVIEW shape file output for channel reaches
y      IDRISI vector output file production
```

These two are straight forward.

```
n      source_hydrograph  calculate subsurface flow hydrograph
n      Identify inner gorges
n      Output inner gorge reaches (IDRISI)
n      Output for fluvial simulation
n      Output for debris flow simulation
```

These five are used only for the simulation model.

```
y      Debris flow inundation hazard
```

A “y” here results in calculation of debris flow inundation hazards and creation of the output raster file hazard_ID.grd

```
n      ARCVIEW shape file output for tributary junctions
```

A “y” here results in creation of a shape file (points) providing information about tributary junctions.

```
n      Valley side slopes
```

A “y” here results in creation of the four valley-wall gradient fields (ValSlp1_R, ValSlp2_R, ValSlp1_L, and ValSlp2_L) in the output shape file.

```
2      Topo index to use for slope stability: 1 = SHALSTAB, 2 = slope
```

This specifies which of the two topographic indexes to use for slope instability with the debris flow inundation model.

Landslide Density

The file dnda.dat provides correspondence between the topographic index of slope stability and landslide density. These values are currently calibrated for landslide inventories in the Oregon Coast Range. Future implementations will provide tools for regional calibration.

BATCH FILES

Batch files can simplify use of these programs when many DEM files need to be examined. An example of a DOS batch file that will run with a Windows prompt is provided below.

```
%1
cd %2
echo %1%2 > tmp1
echo %3 >> tmp1
echo y >> tmp1
echo n >> tmp1
c:\work\source\Bld_grds < tmp1 > log_%3
echo %1%2 > tmp1
echo %3 >> tmp1
echo %3 >> tmp1
echo 4 >> tmp1
c:\work\source\Netrace < tmp1 >> log_%3
del tmp1
del ang_%3.*
del bcont_%3.*
del slope_%3.*
del accum_%3.*
```

```
del Pin_%3.*
del dir_%3.*
del qt_%3.*
del order_%3.*
c:
cd \work\source
```

For this example, three arguments are required: the drive on which the data files are located, the path to the data files, and the file ID. If the batch file is named batch_1.bat and you want to examine data files located on drive e: in the directory \washington\dems\ with ID 1001, then type:

```
batch_1 e: \washington\dems\ 1001
```

Additionally, another batch file that calls batch_1 multiple times can be created. Say, for example, you wanted to run the programs on a series of DEMs, with identifiers 1001 through 1005. You'd then create a second batch file with the following lines:

```
call batch_1 e: \washington\dems\ 1001

call batch_1 e: \washington\dems\ 1002

call batch_1 e: \washington\dems\ 1003

call batch_1 e: \washington\dems\ 1004

call batch_1 e: \washington\dems\ 1005
```

SOURCE LISTINGS

Source files for these programs are available. All code is written in Fortran, with a mix of fixed-format Fortran 77 and free-format Fortran 90. To compile these programs requires linking of object files created from the following files.

Bld_grds

- Big_grds.f90
- Doc_file.f90
- Flood.f90
- Flow.f90
- Nr.f90
- Nrtype.f90
- Nrutil.f0
- Ran.f90
- Sort.f90

Note that the underlined files are from Numerical Recipes (Press, Teukolsky et al. 1996) (www.nr.com).

Newtrace

- Brent.f90
- check_file.f90
- chnslp.for
- convert.f90
- dBASE_record.f90

doc_file.f90
fit.f0
gammln.f90
gammq.f90
gcf.f90
grad.f90
gser.f90
indexx.f90
locate.f90
mainhead.f90
medfit.f90
nr.f90,
nrtype.f90
nrutil.f90
polint.f90
pythag.f90
runout.f90
runtrace.f90
select.f90
svbksb.f90
svdcmp2.f90
svdfit.f90
trace.f90.

Here too the underlined files are from Numerical Recipes.

REFERENCES

Benda, L. (1994). Stochastic Geomorphology in a Humid Mountain Landscape. Geological Sciences. Seattle, WA, University of Washington: 356.

Benda, L. E. and T. W. Cundy (1990). “Predicting deposition of debris flows in mountain channels.” Canadian Geotechnical Journal 27: 409-417.

Dietrich, W. E., D. Bellugi, et al. (2001). Validation of the Shallow Landslide Model, SHALSTAB, for Forest Management. Land Use and Watersheds. M. S. Wigmosta and S. J. Burges. Washington, D.C., American Geophysical Union.

Dietrich, W. E., C. J. Wilson, et al. (1993). “Analysis of erosion thresholds, channel networks, and landscape morphology using a digital terrain model.” Journal of Geology 101: 259-278.

Garbrecht, J. and L. W. Martz (1997). “The assignment of drainage direction over flat surfaces in raster digital elevation models.” Journal of Hydrology 193: 204-213.

Iverson, R. M., S. P. Schilling, et al. (1998). “Objective delineation of lahar-inundation hazard zones.” Geological Society of America Bulletin 110: 972-984.

Jenson, S. K. and J. O. Domingue (1988). “Extracting topographic structure from digital elevation data for geographic information system analysis.” Photogrammetric Engineering and Remote Sensing 54(11): 1593-1600.

Montgomery, D. R. and W. E. Dietrich (1992). "Channel initiation and the problem of landscape scale." Science 255: 826-830.

Montgomery, D. R. and E. Foufoula-Georgiou (1993). "Channel network source representation using digital elevation models." Water Resources Research 29(12): 3925-3934.

Press, W. H., S. A. Teukolsky, et al. (1996). Numerical Recipes in Fortran 90. Cambridge, Cambridge University Press.

Robison, G. E., K. A. Mills, et al. (1999). Storm Impacts and Landslides of 1996: Final Report, Oregon Department of Forestry.

Schilling, S. P. (1998). LAHARZ: GIS programs for automated mapping of lahar-inundation hazard zones. Vancouver, Washington, U.S. Geological Survey.

Tarboton, D. G. (1997). "A new method for the determination of flow directions and upslope areas in grid digital elevation models." Water Resources Research 33(2): 309-319.

Tarboton, D. G. and D. P. Ames (2001). Advances in the mapping of flow networks from digital elevation data. World Water and Environmental Resources Congress, Orlando, Florida.

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

Zevenbergen, L. W. and C. R. Thorne (1987). "Quantitative analysis of land surface topography." Earth Surface Processes and Landforms 12: 47-56.