EVALUATING AND COMPRESSING HYDROLOGY ON SIMPLIFIED TERRAIN

By

Jonathan Muckell

A Thesis Submitted to the Graduate

Faculty of Rensselaer Polytechnic Institute

in Partial Fulfillment of the

Requirements for the Degree of

MASTER OF SCIENCE

Major Subject: COMPUTER SYSTEMS ENGINEERING

Approved:

W. Randolph Franklin, Thesis Adviser

Rensselaer Polytechnic Institute
Troy, New York

April 2008
(For Graduation May 2008)

CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGMENT

This research project would not have been possible without the support of many people. Words can not express my appreciation for these individuals for encouraging me to think innovatively, as well as providing deep insight to the working of the scientific community. I would like to express my gratitude to my advisor, Prof. Franklin who was abundantly helpful and offered invaluable assistance, support and guidance. Deepest gratitude towards Prof. Marcus Andrade and Prof. Barb Cutler for providing great ideas, motivation and insight in the areas of hydrology and computer graphics.

During this work I have collaborated with many colleagues for whom I have great regard, and I wish to extend my warmest thanks to all those who have helped me with my work. Special thanks also to all my graduate friends, especially group members; Prof. Frank Luk , Zhongyi Xie , Metin Inanc, Dan Tracy, and Jake Stookey for sharing the literature and for valuable discussions on terrain representation. I would also like to convey thanks to the Department of Electrical and Computer Systems Engineering and Faculty for providing financial support and laboratory facilities.

Thanks to the various governmental agencies for funding this research, which was supported by NSF grants CCR-0306502 and DMS-0327634, by DARPA/DSO/GeoStar, and by CNPq - the Brazilian Council of Technological and Scientific Development.

ABSTRACT

We present a metric based on the potential energy of water flow to determine the error introduced by terrain simplification algorithms. Typically, terrain compression algorithms seek to minimize RMS (root mean square) and maximum error. These metrics fail to capture whether a reconstructed terrain preserves the drainage network. A quantitative measurement of how accurately a drainage network captures the hydrology is very important for determining the effectiveness of a terrain simplification technique. Having a measurement for testing and comparing different models has the potential to be widely used in numerous applications (floods, erosion, pollutants, etc). In this paper, we first define a metric that maps the reconstructed drainage network onto the original terrain and computes the amount of energy needed for the water to flow. Two novel terrain simplification algorithms are presented that use a targeted compression to preserve the important hydrology features. These methods and other simplification schemes are then evaluated using the potential energy error metric to determine how much hydrology information is lost using the different compression techniques.

1. Introduction

1.1 Trends in Terrain Data and Hydrology

Terrain data is being sampled at ever increasing resolutions over larger geographic areas requiring special compression techniques to manipulate the data. Typically the effectiveness of a terrain compression technique is how well it minimizes the root mean square or the maximum error between the original terrain and the reconstructed geometry [7]. This metric is not always the best choice for preserving hydrological information, since channels and ridges, essential for the calculation of drainage networks [16], might be lost. For example, a scheme which naively interpolates the terrain between two points on opposite sides of a river can flatten the terrain and block flow.

Often measuring the amount of water flow occurs by taking ground truth measurements, where hydrology statistics are determined by direct measurement. This can be expensive, time consuming and requires accessing remote locations. Rapid technological advances are making it possible to have accurate, high resolution elevation data. This provides for a more accurate simulation of hydrology, in ways that were once impractical. In order for this to happen, it is essential that the scientific community has the tools available that can store and manipulate large terrain datasets [1]. Accurate hydrological simulations could allow better understanding of regions at greatest risk of flooding, help minimize the threat of natural disasters and to track and predict the flow of pollutants. This work could also be applied to other flow based models. For instance, instead of water, it could be used to understand threat areas due to volcanic activity. Also, it could be applied to high resolution data for segmentation based on the watersheds.

1.2 Prior Art

Past work has been done for defining a metric for comparing how well a computed drainage compares to the real world drainage [17]. The D8 model can assign flow in one of the eight possible directions. In the SFD (single flow direction) version

of the D8 model the entire amount of flow from each cell is entirely distributed to the lowest adjacent neighbor. This is not the case in the MFD (multi-flow direction) version the flow is fractionally distributed to all the lower adjacent neighbors.

A slightly more sophisticated MFD model is the D∞ model. As the name indicates, flow can travel in an infinite number of directions and is not limited to eight directions. The amount of water leaving each cell is distributed to one or more adjacent cells based on the steepest downward gradient [13].

Another implementation for finding drainage networks are digital elevation model networks or the DEMON model [4]. Instead of modeling flow as a point source that flows to an adjacent neighbor, DEMON captures the flow by contributing and dispersal areas. The motivation for using a method such as DEMON is that the representation allows for flow width to vary over nonplanar topography. However, this can introduce loops and inconsistencies in the hydrology.

Based on the fact that elevation data is only an approximation for the actual terrain, some methods allow for water to flow uphill until spilling over an edge. These flooding methods determine spill points out of every basin. In the Terraflow approach [5, 15], the path of least energy is used to flow uphill until reaching the spill point. The flow runs uphill in situations when there is not an adjacent lower elevation. These methods often keep expanding the drainage networks until they flow off the edge of the terrain. This is because it is assumed that the initial input DEM is prone to collection and sampling errors that cause unrealistic depressions. The main benefits of Terraflow are the ability to avoid dataset issues, obtain long continuous river flow and scalability on massive datasets. The main disadvantages are that this approach may miss realistic drainage basins and poorer performance on non-massive datasets.

Typically for any of the method listed above, the inputs are a DEM (Digital Elevation Model) and a flow accumulation threshold. The outputs are a flow direction grid and a flow accumulation grid. The flow direction grid specifies the direction of flow. The flow accumulation grid is an integer corresponding to the amount of flow and a cell is considered part of the drainage network if its flow accumulation threshold is larger then the threshold value given as an input.

1.3 Overview

The contributions to this research are as follows:

1. A new metric for measuring the amount of hydrology error introduced by a terrain simplification algorithm. The drainage network is computed on the reconstructed terrain and then is mapped onto the original terrain. The amount of potential energy error is computed. Flow traveling uphill will increase the error, while flow traveling downhill will lower the amount of error.

2. Efficient drainage network computation based on a system of linear equations. The resulting drainage often contains longer and more realistic drainage networks then ArcGIS [11] which is typically regarded as the industry standard.

3. Simple, fast and effective computation of the ridge network. Inverting the terrain and running the drainage network provides an approximation of the ridge network. Often compression techniques smooth out ridges. Having an accurate representation of the ridge network can assist compression algorithms and also has applications in siting, path planning and hydrology.

4. Introduction of a new compression method that specifically minimizes hydrology error by using over-determined LaPlacian PDEs.

5. Triangulation algorithms for simplifying terrain are widely used in practice. A variant of a Triangulated Irregular Network called HydroTIN is described that is focused on minimizing the amount of potential energy error.

6. A method for removing insignificant ridges that unrealistically block water flow. These small ridges arise due to dataset and collection errors and are troublesome when executing a drainage network program. By eliminating these regions, longer, more naturally representative rivers result from the drainage network simulation.

2. Measuring Hydrology Error

2.1 Drainage Network Error Metric

Standard metrics for evaluating the effectiveness of terrain simplification algorithms use root mean squared (RMS) and maximum error. These measurements are ineffective when evaluating the loss of drainage network structure. Therefore one of the most important aspects of this paper is to introduce a metric geared towards measuring this error.

It is important to remember that the goal of our hydrology metric is not to compare the reconstructed hydrology against an absolute truth. Hydrology computed on a digital representation may have significant errors due to sampling and data collection inaccuracies. Therefore, our hydrology metric does not compare the reconstructed drainage network versus the original drainage network directly, as with ground-truth methods. Instead, the hydrology metric takes the flow direction grid and the flow accumulation grid from the reconstructed drainage and maps it onto the original DEM (Figure 2.1).

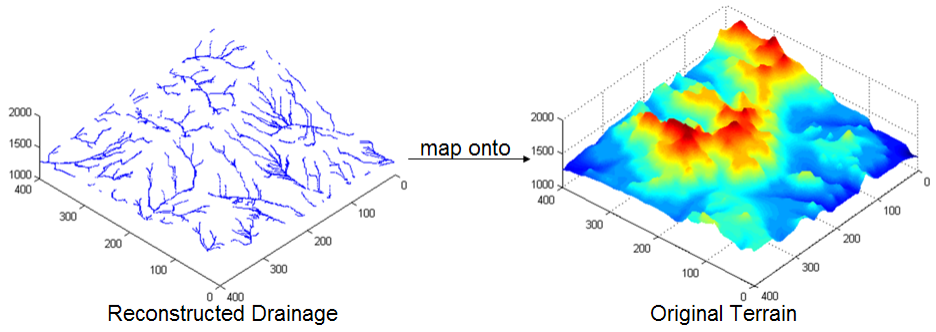

Figure 2.1: To compute the potential energy error, the drainage is computed on the reconstructed terrain. This drainage network is then mapped onto the original terrain. The amount of water flowing uphill and downhill influences the metric.

To compute the accuracy of the drainage network, the gradient, amount of flow contributing cells and whether the flow is traveling uphill or downhill are taken into account. The total downward energy and upward energy is computed as a

summation of the gradient $|(E_i - E_r)|$, where E_i is the original elevation matrix and E_r is the receiving elevation matrix where each cell contains the elevation of the adjacent cell in E_i that is receiving the water flow. The gradient is weighted by the amount of flow (variable W). The final $Error$ is determined as the ratio of the total upward energy divided by the total downward energy.

$$EnergyDown = \sum (E_i - E_r) * W_i$$

$$EnergyUP = \sum (E_r - E_i) * W_i$$

$$Error = \frac{EnergyUP}{EnergyDown}$$

In order to compute the energy error metric the flow is computed on the reconstructed DEM. The error is determined by comparing the flow direction matrix computed on the reconstructed geometry with the elevation matrix from the original DEM. A perfect match would have an energy matrix equal to zero. This would occur if the flow never went uphill, which is the case when using the flow direction grid from the original terrain. Therefore, the closer the metric is to zero, the more accurate the reconstructed drainage network.

2.2 Ridge-River Drainage Calculation

In this work, the drainage network is computed using a standard D8 model [13] based on steepest descent flow. In this implementation each cell flows to the lowest adjacent neighbor and flow is forbidden from traveling uphill. The method is executed on both the original and inverted terrain, and this can be done in parallel. The inverted terrain is derived from the original elevation matrix using the equation below:

$$I_e = Max(E) - E + Min(E) \tag{2.1}$$

where E is the original elevation matrix and I_e is the inverted elevation matrix. The drainage network is computed using E to determine the drainage network and I_e to determine the ridge network. The two networks are combined together and throughout this paper they will be referred to as the ridge-river network, as seen

in Figure 2.2. The process for computing each network is identical except the pre-

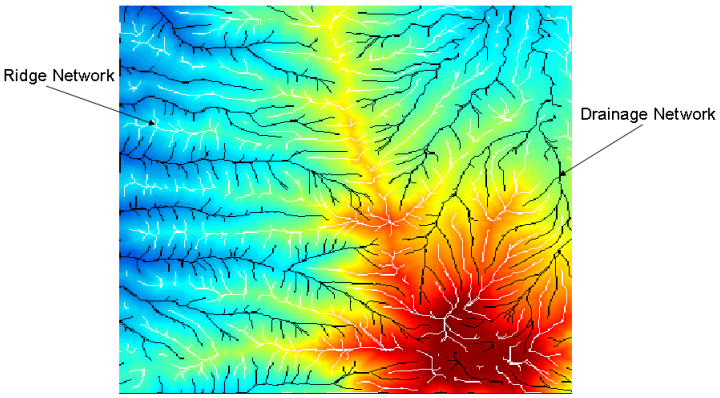

Figure 2.2: The ridge-river network, with rivers in black and ridges in white.

viously defined elevation inversion described above. Figure 2.3 shows our drainage network computation compared to ArcGIS developed by ESRI. Our implementation results in less fragmented and more realistic river networks.

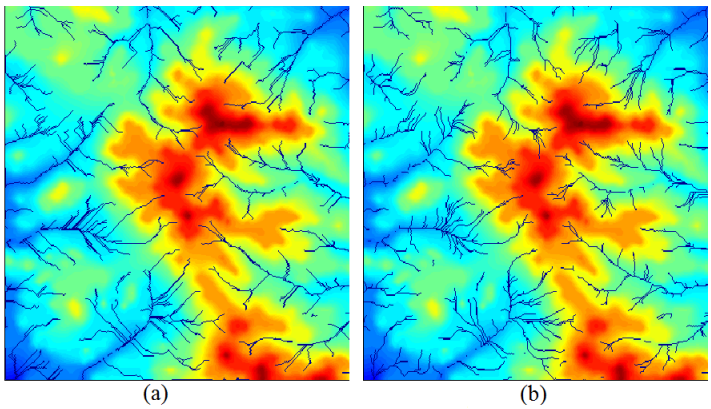

Figure 2.3: (a) Our method for computing the drainage networks compared to (b) ArcGIS. Notice how our method is less fragmented then ArcGIS.

The output of the initial drainage computation is a flow accumulation grid, where each cell contains an integer corresponding to how many other cells contribute flow to that point. Cells above a predefined threshold are considered significant and are added to the drainage and ridge network, which we call the ridge-river network[10]. It is not necessary to store all these cells since they are clustered together and therefore add little value to a point selection compression technique. The

Douglas-Peucker [6] algorithm is used to reduce the number of points required to represent each river segment. The refined points can be stored and further compressed. To reconstruct the terrain we use an implementation of Over-determined Laplacian Partial Differential Equations (ODETLAP) [7] where each point is considered to be the average of its four neighbors, with a subset of the ridge-river points being known.

Different from other methods that use flooding [1], our method computes flow using a system of linear equations $Ax = b$ where x is an unknown N^2 length vector equal to the amount of water accumulation at each cell and b is the initial flow or "rain" at each cell, usually equal to 1. Matrix A is a $N^2 \times N^2$ sparse matrix, where each non-zero element corresponds to cells that contribute flow to each cell. For instance, if cell X_1 receives flow from cell X_2 and X_5, row 1 in matrix A will contain non-zero elements in columns 1, 2 and 5. Therefore the number of non-zero entries in matrix A is bounded by $2N^2$, where N is the size of the $N \times N$ DEM. The upper bound of $2N^2$ is determined since there will be N^2 non-zero entries to load the identity matrix. All other non-zero entries represent flow from one cell to one other cell. There can be at most N^2 additional non-zero elements, since each cell can flow in only one direction. Taking advantage of the sparse nature of matrix A the linear system can be solved efficiently. In Figure 2.4 we show the compute time to initialize and solve the linear system corresponding to the matrix size.

An example of solving the flow accumulation is as follows. Assume we have a trivial 3 by 3 elevation matrix, we the value at each index equals the height of the cell.

$$\begin{pmatrix} 1 & 2 & 4 \\ 3 & 9 & 5 \\ 6 & 7 & 1 \end{pmatrix}$$

Step 1 - Initialize the equations in the form $Ax = b$. Load the identity in matrix A and assume each cell receives 1 unit of rainfall. X_{1-9} equals the amount of flow at each cell.

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \\ X_5 \\ X_6 \\ X_7 \\ X_8 \\ X_9 \end{pmatrix}$$

Step 2 - Determine the direction of flow. Each cell flows to the lowest adjacent neighbor, for simplicity in this example only the 4 orthographic neighbors are considered.

$$\begin{pmatrix} 1 & \leftarrow & 2 & \leftarrow & 4 \\ \uparrow & & \uparrow & & \\ 3 & & 9 & & 5 \\ \uparrow & & & & \downarrow \\ 6 & & 7 & \rightarrow & 1 \end{pmatrix}$$

Step 3 - Add flow information into the equations. For example cell #4, flows all its water into cell #1. So we add 1 in row 1 column 4 in matrix A. Each flow direction (arrow) above creates an additional non-zero entry into matrix A. When accounting for all the flow directions, the linear system will be the following.

$$\begin{pmatrix} 1 & 1 & 0 & \mathbf{1} & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \\ X_5 \\ X_6 \\ X_7 \\ X_8 \\ X_9 \end{pmatrix}$$

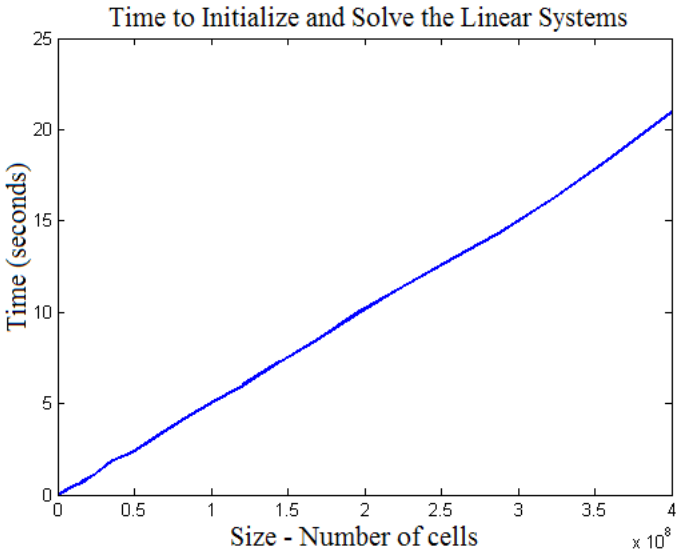

Figure 2.4: Time to initialize and solve the sparse matrix (Matrix A) for small and large datasets.

Step 4 - Solve the equations. This will provide a matrix that contains how much water enters each node called the flow direction grid.

$$\begin{pmatrix} X_1 & X_2 & X_3 \\ X_4 & X_5 & X_6 \\ X_7 & X_8 & X_9 \end{pmatrix} = \begin{pmatrix} 6 & 3 & 1 \\ 2 & 1 & 1 \\ 1 & 1 & 3 \end{pmatrix}$$

The major benefits of this approach is simplicity, scalability and it is consistent (there is never a flow loop). However a significant disadvantage is that this method does not account for flow sampling and dataset inaccuracies that often unrealistically block flow. A method for processing terrain to better handle these terrain errors is described in chapter 5 entitled Hydrology Conditioned Datasets.

2.2.1 Assigning Flow Directions to Plateaus using Connected Components

One additional computation step needs to be performed to deal with an important problem, which is the occurrence of plateaus. These are defined as regions where the flow direction can not be determined based on steepest decent flow.

To deal with these cases, the plateaus are first identified using a very fast variant of the Union-Find algorithm developed by Franklin and Landis [8]. The input is a $3N-2$ by $3N-2$ binary matrix and the output contains a list of components, with each component representing one plateau.

Figure 2.5 shows the plateaus and initial drainage network. Once identifying the flat areas, the cell directions are set using a similar strategy to Terraflow [15], where a breadth-first search assigns directions towards the root or spill point. Spill points are identified as cells in a flat component that contain a nonzero direction. In other words, a cell in the component must have at least one adjacent cell with a smaller elevation. Flat areas that have no spill points are determined to be sinks. The directions of every cell in a sink are assigned to flow to a single point.

After assigning directions to every plateau and sink, the final flow stage can be computed. The linear system of equations is modified to include the directions assigned to the plateaus and sinks. The flow is recomputed and the final flow accumulation grid and flow direction matrix is determined. Figure 2.5 shows our final flow computation with the drainage network and watersheds.

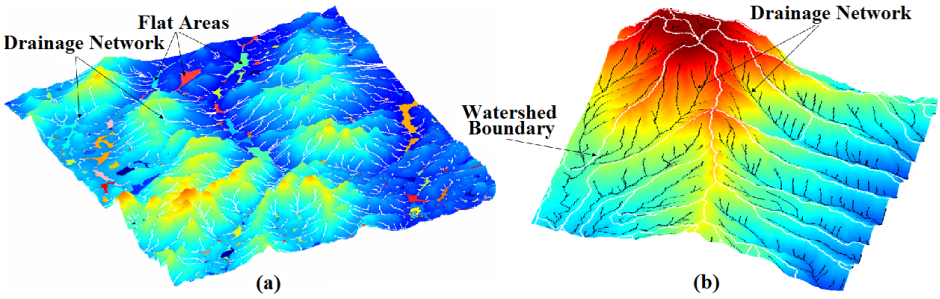

Figure 2.5: (a) Drainage network(white) before handling flat regions. (b) Drainage network (black) and watershed boundary (white) after accounting for flat regions.

Virtually the exact same approach for finding plateaus can be used to determining watersheds. The only difference is that the flow direction matrix is used to initialize the connected components input, instead of the elevation matrix. For each flow direction, we set the connecting binary number from a 1 to 0, exactly the same as we would do for two adjacent elevations of the same height when finding

the plateaus above. In figure 2.6 is a visualization of an isolated watershed found using the connected components program.

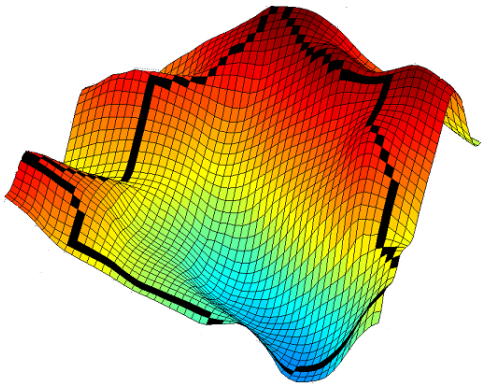

Figure 2.6: The connected components approach can also be used to determine watersheds. This image shows an isolated watershed with the boundary shown in black.

To better illustrate how connected components is useful in determining watersheds and plateaus, a simple example will be shown below. Using the same initial DEM used to solve for the flow accumulation matrix, we will now solve for the watersheds.

First we initial a $2N-1$ by $2N-1$, where N is the size of the $N \times N$ DEM. Also note that for simplicity flow directions are limited to 4 possible orthographic directions. In practice the D8 model is used and flow can travel in diagonal directions and this will also force our connected component matrix to be larger.

$$\begin{pmatrix} 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{pmatrix}$$

Notice that there are nine 0 entries in the matrix above. Each of those zero values directly maps to the elevations in the DEM. Below we see the DEM along with the flow directions.

$$\begin{pmatrix} 1 & \leftarrow & 2 & \leftarrow & 4 \\ \uparrow & & \uparrow & & \\ 3 & & 9 & & 5 \\ \uparrow & & & & \downarrow \\ 6 & & 7 & \rightarrow & 1 \end{pmatrix}$$

For each flow direction above, we set one cell from one to zero.

$$\begin{pmatrix} 0 & \mathbf{0} & 0 & \mathbf{0} & 0 \\ \mathbf{0} & 1 & \mathbf{0} & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ \mathbf{0} & 1 & 1 & 1 & \mathbf{0} \\ 0 & 1 & 0 & \mathbf{0} & 0 \end{pmatrix}$$

Using this as input into the connected components program, the number and location of watersheds is determined. For this simple example there are only 2 watersheds. Hence, the connected components implementation will output 2 components and there locations.

Watersheds can be used to verifying the effectiveness of a terrain simplification algorithm and also for segmentation and environmental analysis.

3. Hydrology-Aware Terrain Simplification

3.1 Terrain Simplification to Preserve the Hydrology

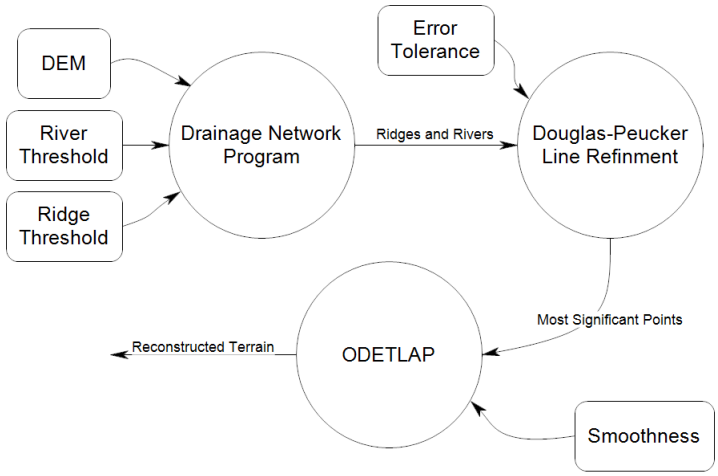

Figure 3.1: Flow chart of the ridge-river technique. Inputs are in boxes and programs in circles.

In Figure 3.1, we show a flow chart describing the ridge-river terrain simplification technique for compressing and uncompressing the hydrology structure of a terrain. Inputs are shown in boxes and programs are shown in circles. The method is based on ODETLAP which reconstructs an approximation of a terrain using a set of points. The recovered terrain achieves a more accurate representation as more points are included.

The basic idea the ODETLAP point selection is to include the most important points that lie on the river and ridge networks. The Drainage Network Program implements the drainage network computation described before and outputs a set of points composing of ridges and rivers. Since these points are clustered together they add little value to a point selection compression technique, thus the Douglas-Peucker algorithm is used to reduce the number of points required to represent each river/ridge segment. This line simplification uses a error tolerance that defines the maximum a simplified line can deviate from the original. These refined pointes are used to represent the terrain.

3.1.1 Approximating Terrain using Over-determined Laplacian PDEs

To reconstruct the terrain from a subset of the original elevation data, we use Over-determined Laplacian Differential Equations (ODETLAP). The input to this method is a compressed subset of points and the output is the reconstructed surface geometry. The Laplacian PDE is extended by adding a new equation to form an over-determined system so that we can control the relative importance of smoothness versus accuracy in the reconstruction. Benefits of the method include the ability to process isolated, scattered elevation points and the fact that reconstructed surface could generate local maxima, which is not possible in the original Laplacian PDE by the maximum principle.

ODETLAP can process not only continuous contour lines but isolated points as well. The surface produced tends to be smooth while preserving high accuracy to the known points. Local maxima are also well preserved. Alternative methods generally sub-sample contours due to limited processing capacity, or ignore isolated points.

Since we are working on single value terrestrial elevation matrix, we have the Laplacian equation for every unknown non-border point.

$$4z_{ij} = z_{i-1,j} + z_{i+1,j} + z_{i,j-1} + z_{i,j+1} \tag{3.1}$$

In terrain modeling this equation has the following limitations:

- The solution of Laplace's equation never has a relative maximum or minimum in the interior of the solution domain, this is called the maximum principle, so local maxima are never generated.
- The generated surface may droop if a set of nested contours is interpolated

To avoid these limitations, an over-determined version of the Laplacian equation is defined as follows: apply the equation (2) to every non-border point, both known and unknown, and a new equation is added for a set S of known points:

$$z_{ij} = h_{ij} \tag{3.2}$$

Where h_{ij} stands for the known elevations of points in S and z_{ij} is the computed elevation for every point, like in equation (2). The system of linear equations is over-determined, i.e., the number of equations exceeds the number of unknown variables, so instead of solving it for an exact solution (which is now impossible), an approximated solution is obtained by setting up a smoothness parameter R that determines the relative importance of accuracy versus smoothness.

3.1.2 ODETLAP Point Selection

In prior work [18], determining which points to input into ODETLAP was based on certain geometric algorithm including Triangulate Irregular Network, Visibility test, Level Set Component that discovers important points which reflect the terrain structure and use our extended Laplacian PDE to approximate the terrain from these points. In this paper the goal is not only to preserve overall terrain structure, but also to ensure that hydrology important features are preserved as well. Our experiments have shown that points on the ridge network and drainage network are the most effective in capturing the hydrology. The ridge-river technique computes both the rivers and ridges, and then simplifies the line network to capture the most significant points.

The drainage and ridge networks are simplified using the Douglas-Peucker[6] line refinement algorithm. This algorithm selects the most significant points need to reconstruct a line within a given error tolerance. This tolerance specifies the maximum distance the line can deviate from the original. Therefore the higher the tolerance, the few points required and the greater the difference between the original network and the reconstructed network. The output from the Douglas-Peucker algorithm is an ordered list of the most significant points needed to reconstruct the line. These points represent our compressed version of the hydrology. As Figure 3.2 illustrates, when the tolerance is set appropriately there is a significant reduction in number of points with the difference in the reconstructed lines being negligible.

To uncompress, we first connect the ordered set of Douglas-Peucker elevation points by using the Bresenham[2] line rasterization algorithm. These points are used as input into ODETLAP which is used to "fill in" the missing data points and is

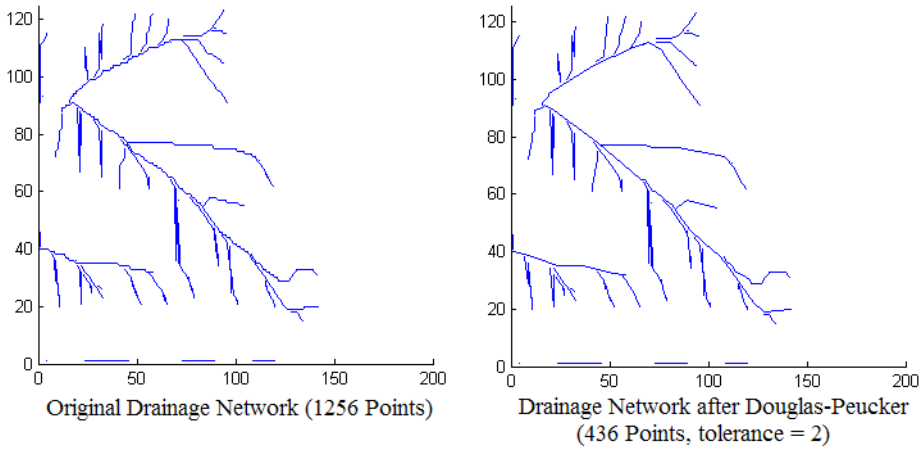

Figure 3.2: Simplifying the original drainage network using Douglas-Peucker. The refined line network is reduced by a factor of 3 with little visible difference.

based on the equations 3.1 and 3.2 specified above.

Our compressed version of the hydrology exists as a subset of elevations from the original DEM, plus the points along the reconstructed line using the Bresenham line rasterization algorithm. All the points are selected along drainage significant features. The reasoning is that we want to capture the most significant points that preserve the hydrology. Therefore, the known points are incorporated into equations 3.3 and 3.4 depending on whether the point came from the river or ridge network respectively. Figure 3.3 shows the ridge-river network.

3.1.3 Recovering Terrain Using a ODETLAP Hydrology Customization

To more accurately capture the structure of the hydrology, the ODETLAP equations are modified for points selected on the ridge-river network. This drastically reduces the amount of error introduced, as shown in Figure 3.3. The modification assume that points on the drainage network are slightly smaller than the average of their 4 neighbors so for river points we can modify equation 3.1 as follows:

$$4z_{ij} = z_{i-1,j} + z_{i+1,j} + z_{i,j-1} + z_{i,j+1} - D_R \tag{3.3}$$

where D_r stands for decrement for the rivers, this variable is an integer corresponding the number of meters the rivers lie below the average of the 4 neighbors. Similarly,

ridge network points are higher then the average of their four neighbors, thus for selected ridge network points, the equation becomes:

$$4z_{ij} = z_{i-1,j} + z_{i+1,j} + z_{i,j-1} + z_{i,j+1} + I_R \tag{3.4}$$

where I_R is an integer corresponding to the increment for the ridges. Experimentation has shown that setting $D_R = I_R = 2$ has been effective. In future work we plan to study how varying this parameter affects the results and investigate ways to automatically select an optimal value.

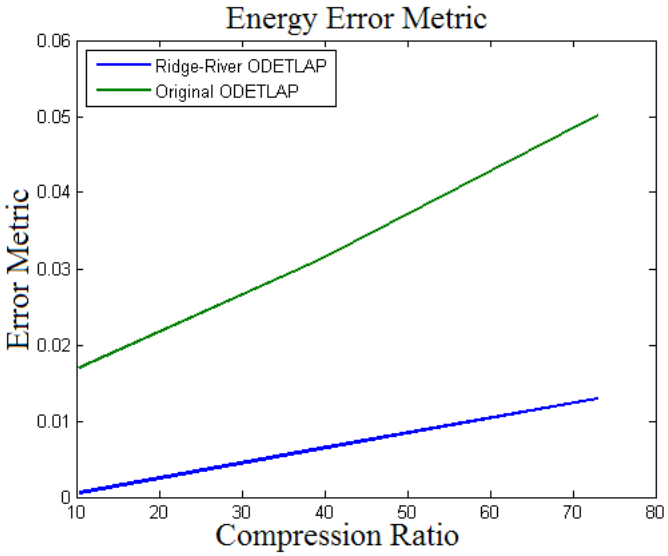

Figure 3.3: Modifying the ODETLAP equations to better represent ridges and rivers has a drastic decrease in the amount of hydrology error. Both plotted lines above use the same set of points.

3.2 Effectiveness of Ridge-Ridge ODETLAP Simplification

The primary focus of this paper has been to describe a metric that accurately captures the amount of error introduced into a reconstructed drainage network. Using this metric, we have been developing an algorithm for achieving high compression ratios without significantly altering the hydrology. The current effectiveness of this approach is shown in the table below. A very common terrain compression is JPEG2000 [3]. This method obtains a low percentage of cells that flow uphill.

This correlates to a fairly low hydrology error. The ridge-river technique described in this paper is effective in achieving high compression ratios with a fairly low error; however, it currently does not consistently beat JPEG2000. We strongly believe that small modifications to the current ridge-river method will allow us to achieve a significantly better hydrology error. We are investigating further modifications to the ODETLAP equations, and to automatically select optimal parameters.

		Ridge-River		JPEG2000		TIN	
DATA	Ratio	% Uphill	ERROR	% Uphill	ERROR	% Uphill	ERROR
hill1	13	2.05%	0.0023	0.12%	0.0020	0.79%	0.0432
	32	3.16%	0.1149	0.18%	0.0030	1.11%	0.0502
	54	2.46%	0.2316	0.24%	0.0082	1.33%	0.0600
hill2	14	0.85%	0.0005	0.21%	0.0010	1.25%	0.0333
	37	1.21%	0.0063	0.31%	0.0017	1.80%	0.0304
	60	1.39%	0.0129	0.46%	0.0047	2.43%	0.0421
hill3	11	2.65%	0.0026	0.10%	0.0059	0.76%	0.0311
	27	4.33%	0.0075	0.11%	0.0051	0.77%	0.0434
	47	2.70%	0.0100	0.13%	0.0161	0.85%	0.0405
mtn1	16	3.75%	0.0267	0.41%	0.0026	3.96%	0.0563
	39	4.96%	0.0530	0.80%	0.0036	5.11%	0.0583
	60	5.91%	0.0611	1.33%	0.0067	6.28%	0.0667
mtn2	16	3.93%	0.0769	0.40%	0.0033	4.42%	0.0748
	38	5.15%	0.1169	0.75%	0.0033	5.72%	0.0874
	59	6.21%	0.1377	1.32%	0.0067	7.09%	0.0904
mtn3	15	3.10%	0.0254	0.40%	0.0015	4.16%	0.0592
	39	4.33%	0.0493	0.78%	0.0027	5.63%	0.0624
	61	5.13%	0.0639	1.40%	0.0050	6.63%	0.0650

Table 3.1: The amount of potential energy error for six 400 by 400 datasets sampled at 30m resolution. The percentage of flow traveling uphill is also shown, along with the compression ratio of each dataset.

Visual inspection of the reconstructed drainage networks correspond to the measurement determined by the potential energy metric. This is observed in Figure 3.4, where the higher error correlates to fragmented and uphill drainage networks. The modular design of our terrain simplification approach allows substituting different algorithms in place of the ones focused on in this paper. For instance, Terraflow or ArcGIS could be used to compute the ridge-river network. Also, a different line simplification technique could be used instead of Douglas-Peucker. This

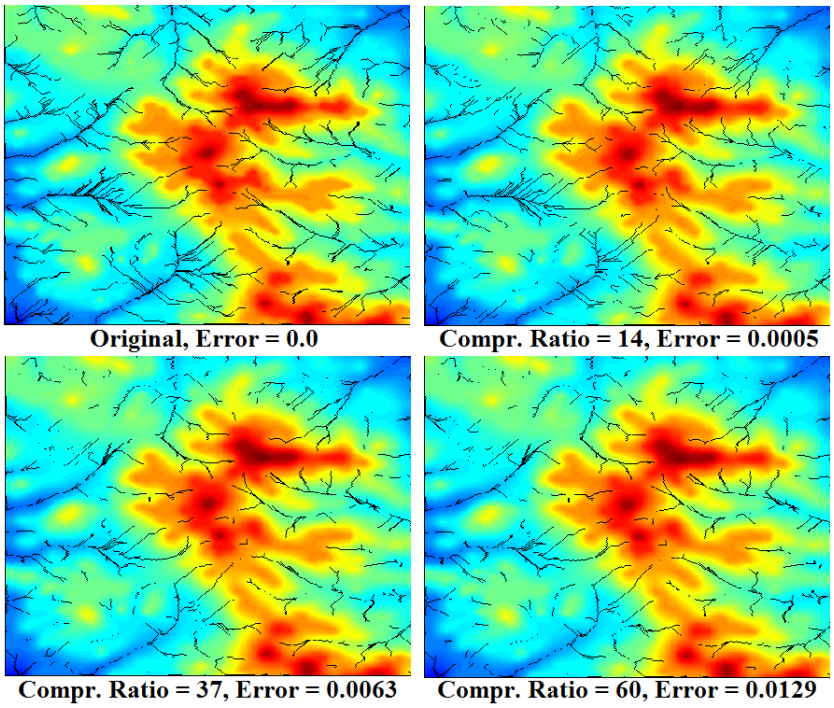

Figure 3.4: The images show the a 400×400 hill2 dataset sampled at 30m resolution and compressed using the ridge-river technique. The color regions represent the elevations with blue being low and red corresponding to high elevation. The black region shows the significant drainage network above the threshold of 100. The higher potential energy error metric correlates with a visible difference in the drainage network. Notice how the high error corresponds to short fragmented drainage networks.

allows modification to fit the specific objectives of the user and application.

Points on the ridges of the terrain, as well as the rivers are important for preserving the hydrology. Rather then use an existing algorithm, we discovered that inverting the terrain and running the drainage network provides a quick, effective method for approximating the ridge network. Once again, this approach can be done with any drainage network program. The ridges are important in terrain compression for extracting and exploiting terrain structure, but also have other GIS applications such as visibility siting, hydrology and edge detection.

4. Hydrology-Aware Triangulation of Terrain

We present a new data structure called HydroTIN for simplifying terrain that captures hydrology significant features using a hydrology-aware Delaunay triangulation. This triangulation preserves the hydrology by using irregular-sized, non-overlapping planes to model regions that flow in a uniform direction. Edges are associated with drainage and ridge networks that incorporate physically-based structure into the model without significant overhead. This allows better compression ratios the standard Triangulated Irregular Networks with higher hydrology accuracy. Standard error metrics such as root mean squared (RMS) and maximum error fail to capture whether a reconstructed terrain accurately captures the hydrology. A hydrology error metric is used to verify our results based on the potential energy required for the reconstructed drainage to flow on the original terrain.

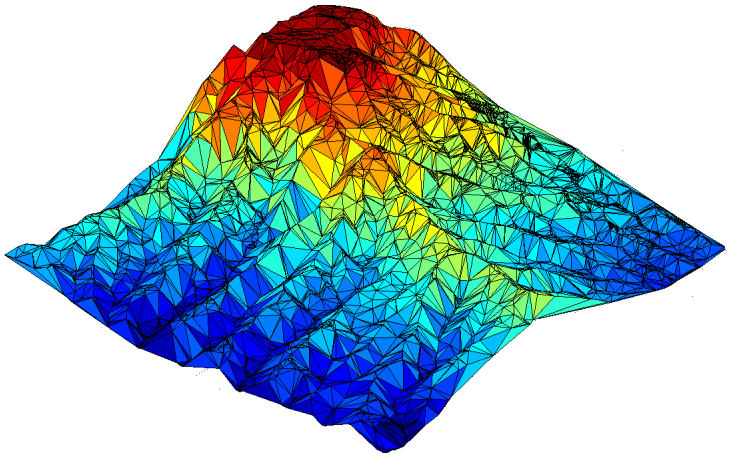

Figure 4.1: HydroTIN is a targeted simplification technique that incorporates the structure of the ridge and rivers and optimized for hydrology preservation.

4.1 Prior GIS Triangulations

Triangulated Irregular Networks or (TIN) [12] is a very popular algorithm for storing a surface for GIS applications. With TIN the surface is stored as a network

of non-overlapping, irregular sized and oriented triangles. This is different then a Digital Elevation Model (DEM) which is a dense raster elevation matrix with the x and y index containing the elevation of z. TIN selects a subset of the n most significant DEM points, these points are then used to construct a Delaunay Triangulation. The Delaunay triangulation is a common computational geometry algorithm that maximizes the minimum angle of all the triangles in the mesh. In our implementation we use the divide and conquer Delaunay triangulation [9] which runs in $O(nlogn)$ time.

Selecting the appropriate points to use for the Delaunay Triangulation is crucial in constructing an accurate representation of the surface geometry without significantly distorting terrain important features. The selection process begins by defining the boundary region of the terrain. This is constructed by dividing terrain into two large triangles that encompasses the entire terrain region. To minimize the maximum error, a point is added one at a time. The point that is the farthest distance from the terrain reconstruction is inserted into the TIN. The terrain is then reconstructed and the average and maximum error is recomputed. This process of inserting more points occurs one at a time until some error threshold is achieved. In this paper, the notion of 'distance' is redefined. Instead of distance being equal to the absolute value of elevation between the TIN and the original DEM, distance is defined using a hydrology error metric. Therefore, instead of minimizing maximum or RMS (root mean squared) error, our HydroTIN algorithm minimizes the amount of drainage network error based on a potential energy error metric. Before selecting points based on their "hydrology distance", the triangle mesh is initial seeded using significant points on the drainage network and ridge network of the terrain.

Numerous methods have been developed for estimating a drainage network from a specified segment of terrain. Based on the fact that elevation data is only an approximation for the actual terrain, some methods allow for water to flow uphill until spilling over an edge. These flooding methods determine spill points out of every basin. In the Terraflow approach [5, 15], the path of least energy is used to flow uphill until reaching the spill point. The flow runs uphill in situations when there is not an adjacent lower elevation. These methods often keep expanding the

drainage networks until they flow off the edge of the terrain. This is because it is assumed that the initial input DEM is prone to collection and sampling errors that cause unrealistic depressions.

Past work has been done for defining a metric for comparing how well a computed drainage compares to the real world drainage [17]. The D8 model can assign flow in one of the eight possible directions. In the SFD (single flow direction) version of the D8 model the entire amount of flow from each cell is entirely distributed to the lowest adjacent neighbor. This is not the case in the MFD (multi-flow direction) version the flow is fractionally distributed to all the lower adjacent neighbors.

A slightly more sophisticated MFD model is the D∞ model. As the name indicates, flow can travel in an infinite number of directions and is not limited to eight directions. The amount of water leaving each cell is distributed to one or more adjacent cells based on the steepest downward gradient [14, 13].

To simplify each curve in the ridge-river network, Douglas-Peucker [6] is used where the reconstructed curve can not deviate more then a predefined tolerance from the original. The output from Douglas-Peucker is a subset of the original points with the first and last point on the curve always existing in the refinement. The number of points used to reconstruct the curve is inversely related to the error tolerance.

4.2 HydroTIN Algorithm

A Delaunay triangulation requires a set of points that will become the vertices of the triangle mesh. Therefore the algorithm for computing HydroTIN selects points considered optimal for preserving the hydrology information. Past work has shown that points on the River and River network are important for hydrology preservation. This Ridge-River network is simplified using the Douglas-Peucker line refinement algorithm to select the most significant points along each ridge-river segment. Figure 3.2 shows this simplification can significantly reduce the number of points required with the difference between the reconstructed line and the refined line being negligible. The refined Ridge-River points are the initial vertex seeding of the triangulation.

Once the initial triangulation is determined, a special case needs to be ad-

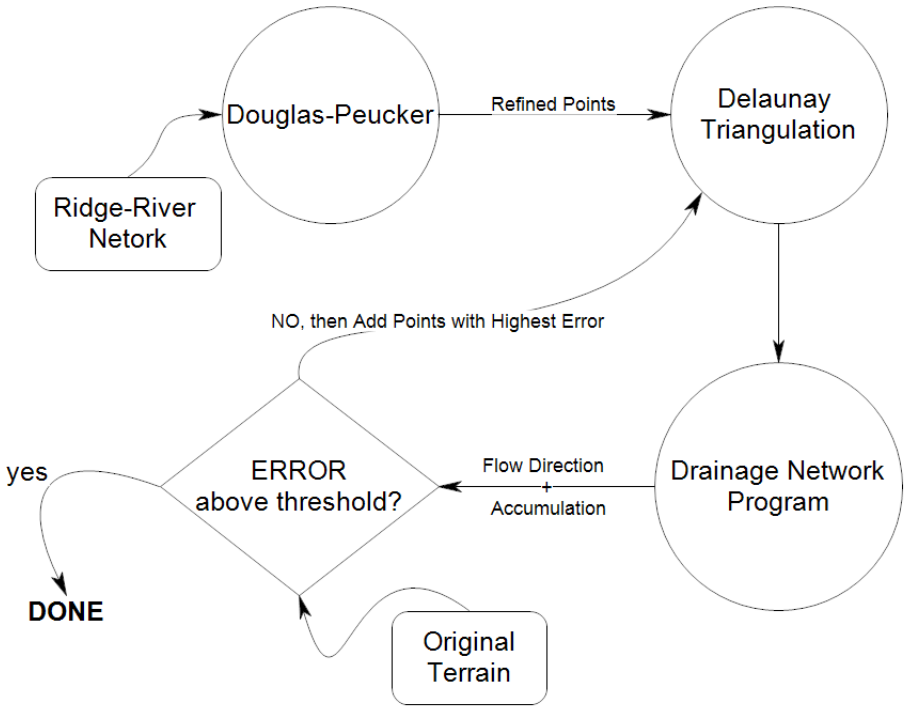

Figure 4.2:

dressed. Ideally, there will be no triangles that are formed that have all three vertices lie on the river network. These triangles cause problems since they flatten out valleys crucial in preserving drainage information. Depending on the input parameters and the terrain dataset, in practice roughly $\frac{1}{5}$ of the triangles will have all 3 vertices on the river network. To correct this problem, the point with the highest elevation within each of these triangles is inserted into the triangulation. Triangles can still form such that they do not contain at least one ridge points or one of the newly insert high elevation points. However, the vast majority are corrected once the Delaunay triangulation is recomputed. Figure 4.2 shows the triangulation after the initial drainage stucture has been encorporated.

At this stage the flow is computed on the reconstructed terrain in order to identify locations at risk for high error. Once identified, the k most points with the highest error are added to the triangulation. The flow is then computed and the process iterates until the amount of error falls below a certain predefined threshold.

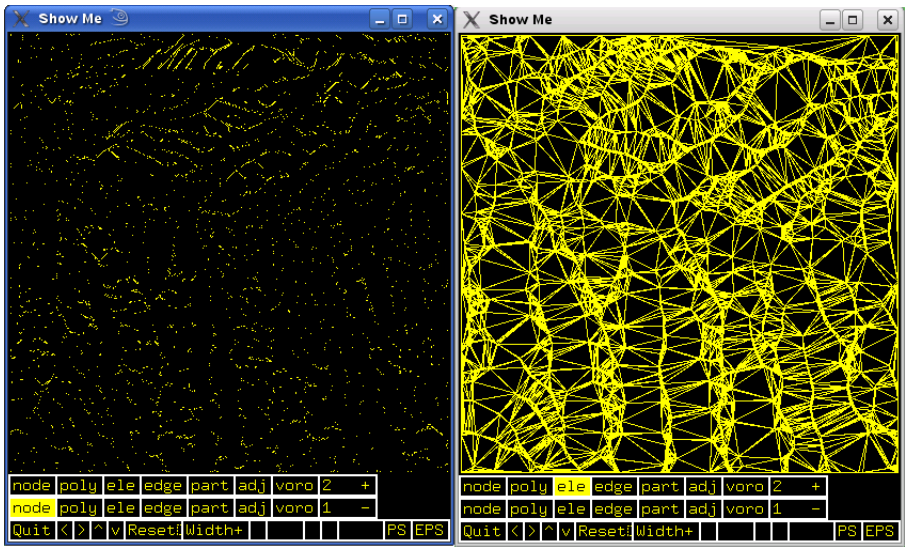

Figure 4.3: Left: Shows ridge-river points refined using Douglas-Peucker. This is the initial "seeding" of the triangulation. Right: Triangulation derived from the nodes on the left. Notice how the edges align with the ridge and river networks.

The optimal value of k is 1, however since the triangulation only effects a local area of the terrain simplification, adding more points a certain distance away from the other points will not effect the reconstruction. Adding more then one point, one can encounter a problem if the points are too close together since the refined points are sometimes clustered. This is because real terrains are mostly continuous so if one point is far away, adjacent points are also likely to be erroneous, and will be selected as well. Because of this, refined points selected by any of our strategies may be redundant in some regions, which is a waste of storage.

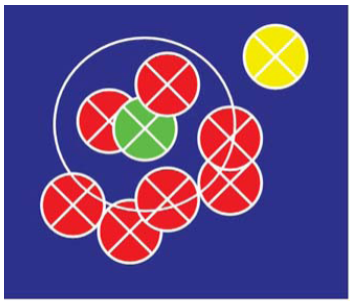

Figure 4.4: Forbidden zone

We perform a check process when adding new refined points: the local neighbor of the new point is checked to see if there is any existing refined points which were added in the same iteration. If yes, this new refined point is discarded and point with the next biggest error is tested until we find desired number of refined points. So as shown in figure 4.4, all potential refined points that are close to an existing refined point(green points) are useless(marked red), and only points that are beyond some distance from green points are selected(marked yellow). By using this method we can add more points per iteration, drastically reducing compute time.

4.3 Results

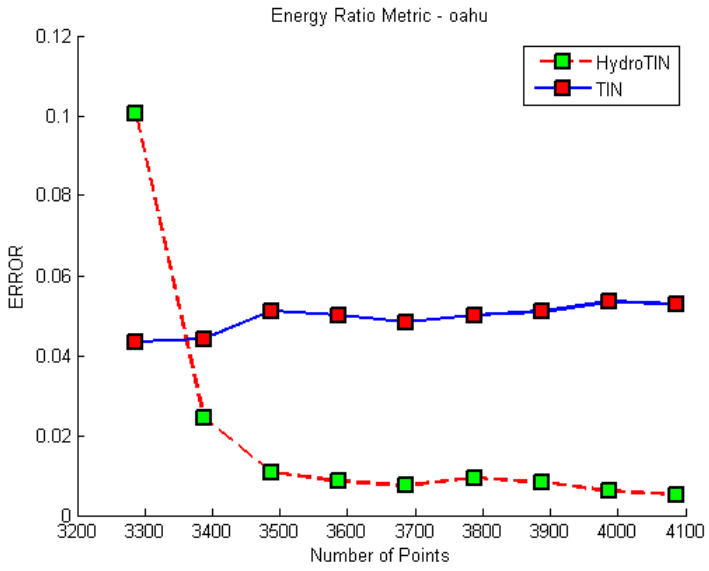

Figure 4.5: Amount of potential energy error determined when mapping the flow direction matrix from the reconstructed terrain onto the original elevation matrix. Error is weighted by the gradient and the amount of flow.

To verify the effectiveness of HydroTIN, our results are compared against terrain computed using a Triangulated Irregular Network (TIN). The specific TIN implementation used in our comparison selects a point that is the farthest away from the reconstructed triangle mesh. Only one point is selected at each iteration.

Figure 4.5 shows an error plot of HydroTIN versus TIN computed using a

400×400 DEM sampled at 30×30 meter resolution of the Hawaiian island of Oahu. For HydroTIN, the initial seeding using the refined ridge-river points contains a relatively high potential energy error. This first stage is important in approximating the core drainage structure, however the error is high due to localized areas of the triangle mesh that have inaccurate slope and contains regions where the flow is traveling uphill when compared to the original. After the first iteration when the 100 points with the highest error are incorporated into the triangulation, the error drops dramatically.

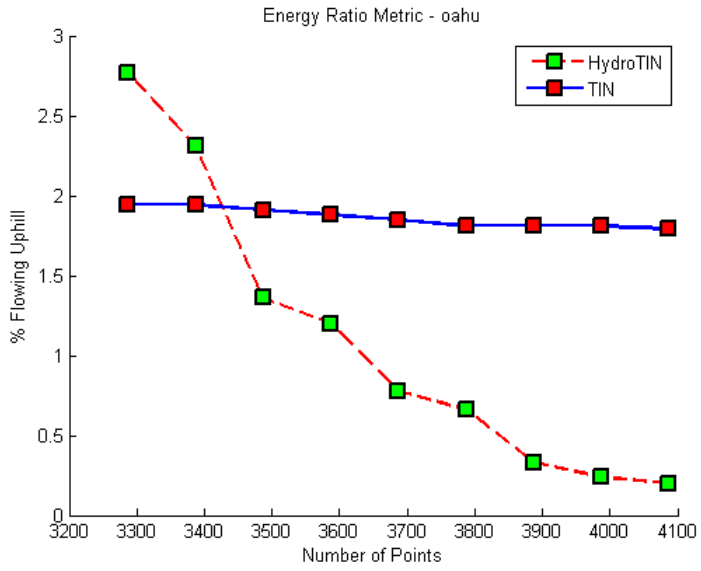

Figure 4.6: Percentage of cells that are flowing uphill when mapping the flow direction matrix obtained from the terrain reconstruction onto the original elevation matrix.

After a couple of iterations HydroTIN will consistently achieve a lower error then TIN, roughly by a factor or 10. Both approaches tend to converge around a certain error value. For TIN, this convergence is nearly immediate since adding new points based on maximum error tends to result in short, fragmented drainage networks that are not representative of naturally occurring flow. These fragmented rivers tend to have a relatively small number of cells contributing in the watershed.

Not surprisingly, there is a high correlation between the error and the number of points that are flowing uphill. In figure 4.6 the relationship between the number

of points and the percentage of points flowing uphill is shown. As more points are included the numbers of cells flowing uphill are reduced for both implementations. However, only a very small decrease is observed for TIN. In contrast, HydroTIN consistently drops the number of upward flowing points. In the last iteration, the number of cells flowing up hill for HydroTIN is a mere 319 out of $160,000$, or 0.2%. For TIN, $2,864$ cells flow uphill or 1.79%.

A user defined input determines the tradeoff between compression size and the amount of potential energy error. The program will continue to iterate until the error is below a user defined threshold. Figure 4.1 is a visualization of HydroTIN which is stored by compressing less then 3500 points and achieves an error of 0.01. The edges of the triangles tend to follow the ridge and river networks preserves drainage structure and preventing significant blocking of water flow. After two iterations of minimizing the amount of potential energy error drops by a factor of 10 by adding just 200 points.

5. Hydrolgy-Conditioned Datasets

5.1 Introduction

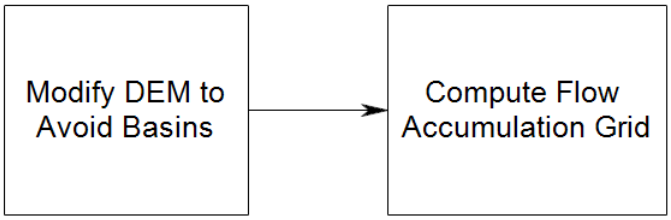

Figure 5.1: Low coupled, two stage process for computing drainage networks to avoid sampling and data collection issues.

Drainage network algorithms often provide fragmented river segments because of small sinks in the approximated terrain. Depressions of a tiny size such as only one cell can be fixed using a simple median filter. However, the challenge of fixing much larger depressions is a difficult problem to overcome. Approaches such as Terraflow [15] will find the path of lowest energy uphill out of these depressions. This yields long continuous river segments. However, this does not accurately model the physical properties of water flow. This section describes a method for modifying the initial Digital Elevation Model (DEM) such that the resulting drainage network will have small insignificant ridges removed from the DEM (Figure 5.2). This allows water to flow passed these areas using a standard physically-based drainage algorithm such as steepest-decent flow.

5.2 Results

Our reconstructed terrain captures the important aspects of the drainage network while still achieving a high compression rate. The reconstruction also has a more natural and realistic representation of the original hydrology because small insignificant ridges have been removed in the point selection process. This results in larger, fewer watersheds. The recomputed drainage network is also captured accurately, besides the small tributaries which aren't considered of high importance. We

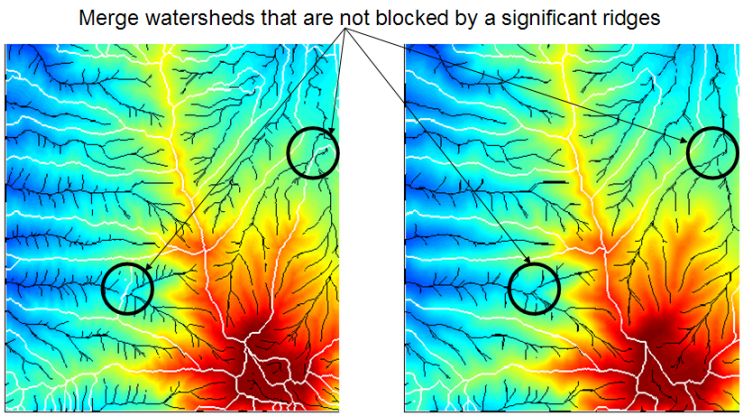

Figure 5.2: LEFT: Original Terrain with watersheds (white) and drainage network (black). The flow here is blocked by a small ridge that occurs due to sampling and data collection errors. RIGHT: Flow passes through small insignificant barriers. Allowing larger and more realistic watersheds and drainage networks

can also store the compressed version using far fewer points then the original DEM. The user can define the level of detail and hence the number of points by adjusting the tolerance level for the Douglas-Peucker algorithm.

Figure 5.3 shows the drainage network computed on four instances of a 400x400 elevation matrix representing a segment of the Hawaiian island of Oahu. In (a) the hydrology was computed on the original elevation matrix. (b) and (c) correspond to hydrology computed on the reconstructed terrain using ODETLAP, where in (b) the points were selected using our ridge-river technique described above, and in (c) using the original ODETLAP method (in each iteration the k "farthest points" were included). In (d) the hydrology is computed on a terrain is recovered from a lossy JPEG2000 compression. All the reconstructions have a similar RMS error of about 8.5.

The first results showed that the hydrology consistency is better preserved on terrain recovered based on the ridge-river point selection method than using original ODETLAP point selection and JPEG2000. Another interesting investigation would be to use the drainage network as a model of natural terrain formation. This could be used to extract structure from the terrain for segmentation and division, allowing for better compression.

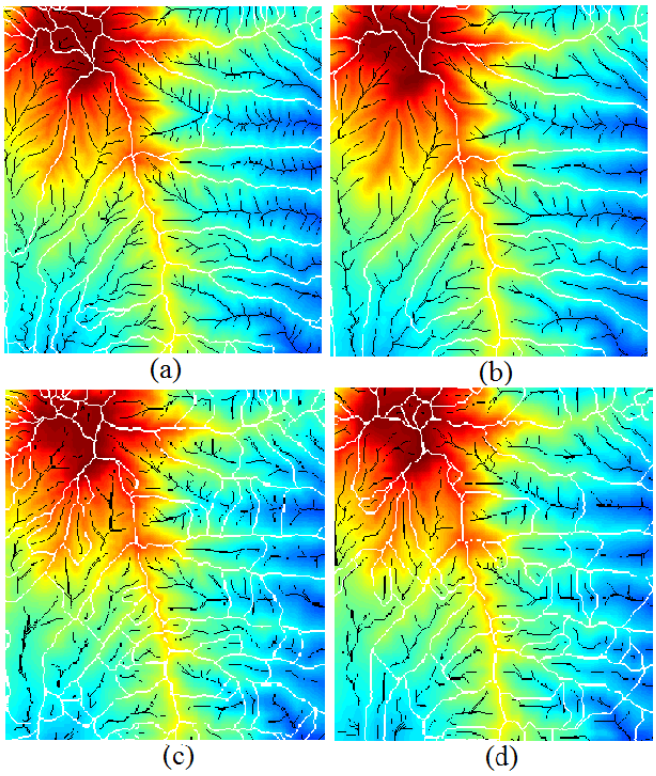

Figure 5.3: The watersheds(white) and drainage networks(black) on the original and recovered terrains.

CONCULSION

The potential energy metric introduced in this paper provides a quantitative measurement of the amount of error introduced by a terrain simplification technique. This value is reflective of the visible examination of the drainage networks, with higher error corresponding to fragmented and unrealistic flow directions (flow traveling uphill).

The original DEM is an approximation of the real world terrain surface and not the complete truth, due to dataset and sampling errors. Therefore it would be inaccurate to compare one drainage network computed on the original DEM versus the drainage network computed on the reconstructed surface. Instead, the drainage network is computed on the reconstructed surface and compared against the original terrain. Flow can travel in different directions then the original drainage network, yet contain a low error metric if the flow directions are reasonable. Standard error metrics such as standard metrics such as root mean squared error and maximum error are ineffective in evaluating the amount of error introduced, as they do not take into account important hydrology features.

As more terrain is being sampled at ever increasing resolutions, it becomes more important to be able store and manipulate large elevation datasets and evaluate the amount of error introduced by lossy compression. However, current techniques for compressing these datasets lose important information that is essential for running operations on the reconstructed geometry with reliable results. Understanding how compression affects important terrain structure, such as hydrology, allows the GIS community to understand how a compression technique affects the drainage accuracy of the reconstructed terrain. Our targeted compression technique has the goal of minimizing the amount of potential energy error, thus allowing for high compression ratios with minimal loss of hydrology information, while at the expense of other terrain structure. The net result of this work is a compression scheme and error evaluation metric with applications including flooding, erosion, sanitation, and environmental protection.

REFERENCES

[1] L. Arge, J. S. Chase, P. N. Halpin, L. Toma, J. S. Vitter, D. Urban, and R. Wickremesinghe. Efficient flow computation on massive grid terrain datasets. *GeoInform.*, 7(4):283–313, 2003.

[2] J Bresenham. A linear algorithm for incremental digital display of circular arcs. *Commun. ACM*, 20(2):100–106, 1977.

[3] JPEG Committee. Jpeg 2000, [Online; accessed 14-March-2008].

[4] M. C. Costa-Cabral and S. J. Burges. Digital elevation model networks (DEMON): A model of flow over hillslopes for computation of contributing and dispersal areas. *Water Resources Research*, 30:1681–1692, 1994.

[5] A. Danner, T. Molhave, K. Yi, P. K. Agarwal, L. Arge, and H. Mitasova. Terrastream: from elevation data to watershed hierarchies. In *GIS '07: Proceedings of the 15th annual ACM international symposium on Advances in geographic information systems*, pages 1–8, New York, NY, USA, 2007. ACM.

[6] D.H. Douglas and T.K. Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. In *The Canadian Cartographer.*, volume 2, pages 112–122, 1973.

[7] W. R. Franklin, M. Inanc, Z. Xie, D. M. Tracy, B. Cutler, M. A. Andrade, and F. Luk. Smugglers and border guards - the geostar project at rpi. In *15th ACM International Symposium on Advances in Geographic Information Systems (ACM GIS 2007*, Seattle, WA, USA, November 2007.

[8] W. R. Franklin and E. Landis. Connected components on 1000x1000x1000 datasets. In *16th Fall Workshop in Computational Geometry*, Smith College, Northampton, MA, 1011 Nov 2006. (abstracts only).

[9] D. T. Lee and B. J. Schachter. Two algorithms for constructing a delaunay triangulation. In *International Journal of Parallel Programming*, volume 9, pages 219–242. Springer Netherlands, 1980.

[10] J. Muckell, M. Andrade, W. R. Franklin, B. Cutler, M. Inanc, Z. Xie, and D. M. Tracy. Drainage network and watershed reconstruction on simplified terrain. In *17th Fall Workshop in Computational Geometry*, IBM T.J. Watson Research Center, Hawthorne, New York, 9-10 Nov 2007. (abstracts only).

[11] T. Ormsby, E. Napoleon, R. Burke, and C. Groessl. *Getting to Know Arcgis Desktop: The Basics of ArcView, Arceditor, and Arcinfo*. ESRI Press, 2001.

[12] T. K. Peucker. Digital representation of three-dimensional surfaces by triangulated irregular networks (tin), 1976.

[13] D. G. Tarboton. A new method for the determination of flow directions and upslope areas in grid digital elevation models. In *Water Resour. Res.*, volume 2, pages 309–320, 1997.

[14] D. G. Tarboton and D. P. Ames. Advances in the mapping of flow networks from digital elevation data.

[15] L. Toma, R. Wickremesinghe, L. Arge, J. S. Chase, J. S. Vitter, P. N. Halpin, and D. Urban. Flow computation on massive grids. In *ACM-GIS*, pages 82–87, 2001.

[16] J. V. Vogt, R. Colombo, and F. Bertolo. Deriving drainage networks and catchment boundaries: a new methodology combining digital elevation data and environmental characteristics. *Geomorph.*, 53((3-4)):281–298, 2003.

[17] J. P. Walker and G. R. Willgoose. On the effect of digital elevation model accuracy on hydrology and geomorphology. *Water Resources Research*, 35:2259–2268, 1999.

[18] Z. Xie, W. R. Franklin, B. Cutler, M. Andrade, and M. Inanc. Surface compression using over-determined laplacian approximation. In *SPIE 2007*, August 2007.