

ANALYSIS OF TOPOGRAPHY AND SHEETING JOINTS
AT THE FLETCHER GRANITE QUARRY, MASSACHUSETTS

A THESIS SUBMITTED TO
THE GEOLOGY AND GEOPHYSICS DEPARTMENT
BACHELORS OF SCIENCE IN GEOLOGY AND GEOPHYSICS
AUGUST 2015

By
Zachary T. Olds

Thesis Advisor
Stephen J. Martel

I certify that I have read this thesis and that, in my opinion, it is satisfactory in scope and quality.

THESIS ADVISOR

A handwritten signature in cursive script, reading "Stephen J. Martel", is written over a horizontal line.

Stephen J. Martel
Department of Geology and Geophysics

Acknowledgements

I would like to thank everyone who helped me make this thesis possible. I sincerely appreciate all of those who took the time out of their day to help me with this project. Thank you to Dr. Steve Martel, my thesis advisor, who helped me every step of the way and took the time to answer even the most inane questions. Without your guidance, counsel, and expertise this project would have never come to fruition. Thank you to Dr. Paul Wessel who helped troubleshoot much of the MATLAB code used for interpolation, and even donated his own code to try to help improve upon the results. You never balked at any questions Steve and I came to you with, and there were many. Thank you to Dr. Scott Rowland who offered help and advice to improve upon the digitization of a very old map, and who also taught me the difference between fluff and scientific writing. Without you this thesis would be a lot more colorful, and a lot less quantitative. I would like to thank the staff at UROP who saw potential in this thesis, and funded my efforts. Thank you to all the professors at the University of Hawaii over the years that taught and shared their passion for geology with me. Finally I would like to thank my family for always being there, and supporting me even when I decide I want to move to the middle of the Pacific Ocean.

Abstract

Sheeting joints are rock fractures that open parallel to the topographic surface at shallow depths. They affect landscape development, groundwater systems, and slope stability. There are many hypotheses for how they form; the most common attributes them to the removal of vertical overburden through erosion. An alternative hypothesis posits that they open as a result of tension induced by high surface-parallel compressive stresses acting along convex topographic surfaces. This thesis tests both hypotheses using an approach based on curvature analysis that classifies topography as one of six topographic shapes: bowls, valleys, planes, saddles, ridges, and domes. The Fletcher Granite Quarry in Massachusetts was selected to test these hypotheses because well-documented sheeting joints are present, a detailed map indicates the locations of 50 sheeting joints, and stress measurements indicate high levels of horizontal compressive stress. A map from 1943 was digitized to obtain the data required to analyze the topographic curvature, as the topography in the region has been substantially altered due to human activity since the locations of sheeting joints in the area were recorded. The curvature analysis shows that 68.2% of the sheeting joints at Fletcher Granite Quarry were located on topographic surfaces that were convex in at least one direction at the time the joints were mapped. This supports the hypothesis of high surface-parallel compressive stresses as the primary mechanism for the formation of sheeting joints.

Table of Contents

Acknowledgements	iii
Abstract.....	iv
List of Tables	vi
List of Charts, Graphs, Figures, Illustrations, Plates, and Maps	vii
Chapter 1. Introduction	1
1.1 Sheeting Joints.....	1
1.2 Purpose	2
1.3 Hypotheses for Sheeting Joints	3
1.3.1 Lateral Compressive Stress Hypothesis	3
1.3.2 Relief of Overburden Hypothesis	6
1.4 Curvatures	6
1.5 Calculating Curvature.....	7
Chapter 2. Methods	10
Chapter 3. Results.....	15
Chapter 4. Discussion and Conclusion	17
4.1 Discussion.....	17
4.2 Conclusions	18
4.3 Future Research	18
Appendix: Codes	19
References	40

List of Tables

1. Topographic Shapes.....	8
2. Distribution of Sheeting Joints at Fletcher Granite Quarry.....	16

List of Charts, Graphs, Figures, Illustrations, Plates, and Maps

1. Sheeting Joints at Fletcher Granite Quarry.....	1
2. Cross Section of Convex Element.....	4
3. Curvature Reference Frame.....	4
4. Traction Free Surface Plot.....	5
5. Relief of Overburden Cartoon.....	6
6. Jahns (1943) Map of Fletcher Granite Quarry.....	11
7. Modified Jahns (1943) Map Showing Sheeting Joints.....	11
8. Interpolation Function Contour Maps.....	12
9. Low Amplitude Vertical Artifacts.....	13
10. Course Grid Pixilation.....	13
11. Effects of Tolerance on Curvature.....	14
12. Final Curvature Map using 'splin2d'.....	15
13. Final Curvature Map using 'griddata'.....	15
14. Topographic Shape Histograms.....	16

Chapter 1. Introduction

The topographic surface of the Earth affects many Earth processes, which in turn shape the world we live in. One such process is the formation of rock fractures that occur at or near the topographic surface, known as sheeting joints. The formation of these surficial fractures can trigger mass wasting events (e.g., Stock et al., 2012), affect landscape development, and control shallow groundwater systems. In this thesis I address how sheeting joints relate to the topographic surface at the Fletcher Granite Quarry, Massachusetts. This interaction can be quantified by evaluating the principal curvatures of the surface (Gauss, 1827); these curvatures will describe the shape of a surface at any point on the surface. Although the region of interest at the Fletcher Granite Quarry is relatively small, $\sim 7 \text{ km}^2$, the methods used in this thesis can be applied to any location where sheeting joints are found.

1.1 Sheeting Joints

Sheeting joints, or exfoliation joints, are surficial fractures, or joints, that form sub-parallel to the topographic surface at shallow depths. They appear similar to the layers of an onion (Fig. 1). Individual sheets of rock between these joints are characteristically lenticular or tabular in form, and can extend in a plane for at least 200 meters (Jahns, 1943; Martel, 2006). Sheeting joints can occur within a meter of the topographic surface and can form at depths greater than 100 meters (Martel, 2006; Jahns, 1943). With depth, joint spacing increases and curvature decreases, creating sheets that are progressively thicker, more horizontal, and less irregular. Sheeting joints most commonly occur where the topography is convex, but have also been

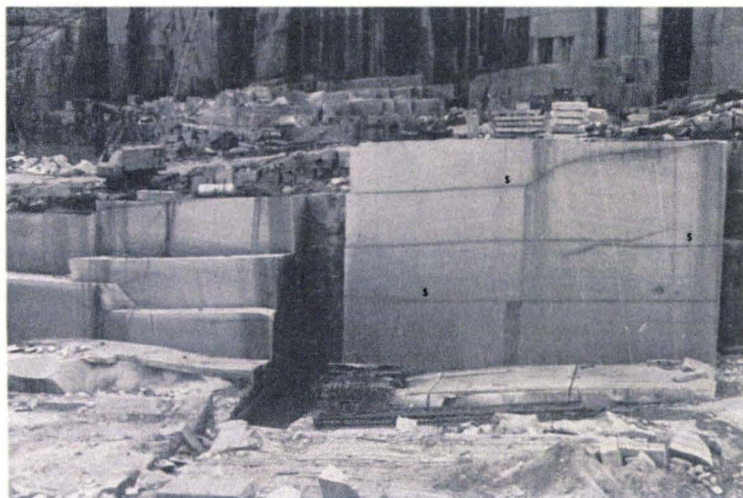


Figure 1. Photograph from the bottom of the Fletcher Granite Quarry, Massachusetts. Sheeting joints are nearly horizontal because they are deep below the original surface, and are labeled with an “S.” (Holzhausen, 1989).

observed in areas where the topography is concave in at least one direction (Martel, 2006). Sheeting joints are entirely independent of structures within the country rock, and transect foliation, dikes, and xenoliths (Jahns, 1943). Observations of sheeting joints favor an open-mode origin, as they typically show no sign of slip (Holzhausen, 1989). Sheeting joints typically have walls that are in contact but they can have apertures of several centimeters (e.g., Matthes, 1930). The ends of sheeting joints commonly end in a pattern of echelon fractures (Martel, 2006). Sheeting joints also exhibit hackles, plumose structures, elliptical ribs, and other structures (Holzhausen, 1989) associated with opening-mode fractures. Sheeting joints are one of the youngest structures in granitic country rock; and commonly mimic topography that formed within the last few million years. Sheeting joints have also developed during mining operations (Holzhausen, 1989). High regional horizontal compressive stresses have been measured or deduced in many locations where sheeting joints have been observed, including at sites inspected by Jahns (1943) at or near the Fletcher Granite Quarry. The mean horizontal compressive stress at the quarry was later measured by Holzhausen (1989) as ~ 31 MPa at an average depth of 60 meters. The presence of high lateral compressive stresses in many locations with sheeting joints indicates that these stresses contribute to sheeting joint formation.

1.2 Purpose

A clearer understanding of how sheeting joints form should be helpful in establishing their role in developing landscapes, as well as for reducing risks to people and property. Yosemite National Park, for example, could benefit in both of these areas. Sheeting joints dominate the landscape throughout the park, which is home to some of the most iconic scenery in the United States. In the past, sheeting joints weakened the bedrock, allowing glaciers and flowing water to shape the solid granite. Now they decorate landmarks such as Half Dome and Matthes Crest (Huber, 1989). Sheeting joints are also a major contributor to rockfalls in the park (Stock et al., 2012) posing a hazard to visitors. The National Park Service is actively involved in rockfall research to mitigate these hazards (Stock and Collins, 2014). General research on sheeting joints can help mitigate hazards associated with rockfalls by facilitating the identification of areas that might be prone to rockfall.

Although much research has been done on sheeting joints, their exact cause and the mechanics behind their formation is still foggy. The purpose of this thesis is to test the hypotheses of Martel (2006, 2011a) and Jahns (1943) using quantitative data in order to help establish the likely mechanism for the formation of sheeting joints at the Fletcher Granite Quarry.

1.3 Hypotheses for Sheeting Joints

Sheeting joints were first described over two centuries ago, and numerous hypotheses attempt to describe their formation. The most widely cited states that relief of overburden is the primary mechanism that causes sheeting joints. Jahns (1943) invoked this hypothesis to account for the formation of sheeting joints at the Fletcher Granite Quarry. However, removal of vertical confining pressure alone cannot produce a tension that would open sheeting joints (Martel, 2006). A theoretical model for sheeting joint formation put forth by Martel (2006, 2011a) explains how high horizontal compressive stresses can induce a vertical tension that would promote sheeting joints. Mechanically, this hypothesis relies on stresses within the rock parallel to the surface interacting with the topography to create a tension large enough to open up a fracture in the rock (Martel, 2011a). Hypotheses such as the cooling of granite, solar insolation, chemical alteration, and fire and frost have been largely discounted (Farmin, 1937; Jahns, 1943). This thesis deals with two main hypotheses that describe the formation of sheeting joints: (a) lateral compressive stress acting along a convex surface, and (b) relief of overburden by erosion.

1.3.1 Lateral Compressive Stress Hypothesis

The lateral compressive stress hypothesis posits that the conditions necessary for the formation of sheeting joints can be achieved if the following inequality is satisfied (Martel, 2011a):

$$\sigma_{11}k_1 + \sigma_{22}k_2 > \rho g \cos\beta, \quad (1)$$

where σ_{11} and σ_{22} are tensile (positive) or compressive (negative) stresses parallel to the topographic surface and in the directions of the principal curvatures k_1 and k_2 respectively, ρ is the rock density, g is gravity, and β is the slope of the surface. This inequality can be used to quantitatively test the compressive stress hypothesis. If the sum of the products of the stress terms and the curvatures is greater than the stress

gradient due to gravity, then sheeting joints should be abundant. If not, then sheeting joints should be sparse or entirely absent (Martel, 2011a). The mechanism proposed by Martel (2011a) can be demonstrated by a free body diagram of a thin convex element (Fig. 2). The element can be thought of as a representation of the outermost surface of an outcrop of rock. The compressional stresses acting on the rock parallel to the convex surface produce an outward radial force within the rock. Preventing this layer from separating from the outcrop is the tensile strength of the rock, which is relatively small, as well as gravity. If the radial outward force overcomes radial inward force due to gravity, then the system will no longer be in equilibrium and the surface layer of rock will separate at the base of the layer; creating a sheeting joint (Martel, 2011a). The mechanics of this approach are based on the equation of static equilibrium and utilize a local reference frame built upon orthogonal curvilinear coordinates (Fig. 3) (Martel, 2011a). The static equilibrium equation relates stresses acting on a body of any rheology, and applying the components of our problem to it produces the equation (see Martel, 2011a for a full derivation):

$$\frac{\partial T}{\partial z} \Big|_{z=0} = \phi = \sigma_{11}k_1 + \sigma_{22}k_2 - \rho g \cos \beta, \quad (2)$$

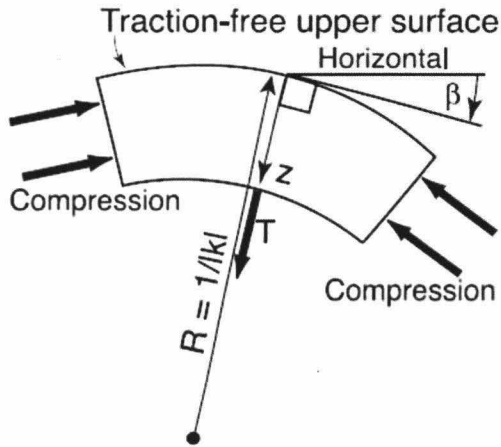


Figure 2. Cross section of a thin convex element with a traction free surface, with tensile stress (T) at the base. Bold arrows indicate tractions acting on the element. R is the radius of curvature of the surface of the element, β is the tangent slope. (Martel, 2011a).

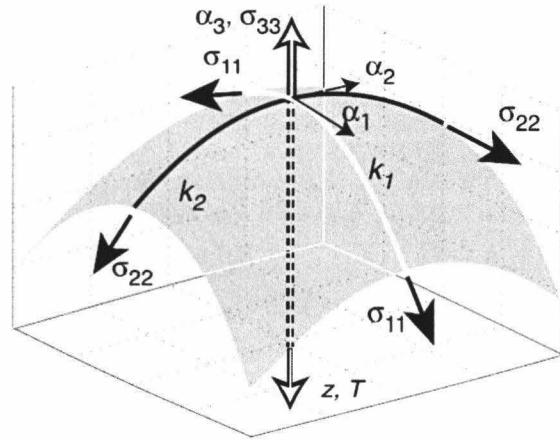


Figure 3. Surface showing the local curvilinear reference frame. The α_1 , α_2 , and α_3 directions correspond to the σ_{11} , σ_{22} , and σ_{33} stress components, respectively. The principal curvatures k_1 and k_2 are parallel to α_1 and α_2 . (Martel, 2011a).

where T is the stress component acting in the z direction, normal to the tangent plane at the surface. At the surface, where z is equal to zero, equation (2) yields the slope of the tension vs. depth curve (Fig. 4), shown as ϕ . Equation (2) shows the normal traction gradient at the surface

of an outcrop. If ϕ is greater than zero then a tension perpendicular to the surface exists at

shallow depths, this is a sufficient criterion for the nucleation of sheeting joints. For this criterion to be satisfied several conditions must exist: (a) the surface-parallel stress terms must be sufficiently compressive (negative), (b) the curvature terms must be sufficiently convex (negative), and (c) the sum of the stress-curvature products must be greater than the product of the unit weight of rock, $\sim 2.64 \times 10^4 \text{ Pa}\cdot\text{m}^{-1}$ for Fletcher granite (FletcherGranite.com, 2015), and the cosine of the surficial slope (Martel, 2011a). This is consistent with the prevalence of sheeting joints in convex topography such as domes and ridges, as ϕ would be positive with a sufficient compressive stress parallel to the surface. It is also consistent with the sparsity of sheeting joints in concave topography such as valleys and bowls, where the product of the curvature of a concave (positive) surface and a surface-parallel compressive (negative) stress would cause ϕ to be negative. In the case of planar topography, the two curvature terms would be zero, causing ϕ to be negative. The topography of a saddle is a unique case; it is concave and convex in perpendicular directions. As many curvature-stress products are possible, the magnitudes of the product terms will dictate if ϕ is greater than zero, and hence if sheeting joints will likely form below a saddle. The lateral compressional stress hypothesis requires topography to be convex in at least one direction for sheeting joints to nucleate. As such, for the hypothesis to be validated sheeting joints must be concentrated on convex surfaces (e.g., domes, ridges, and saddles).

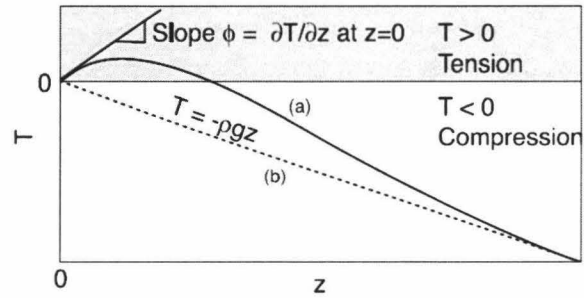


Figure 4. Plot showing tension below a traction free surface (T) as a function of depth normal to the surface (z). The slope at $z = 0$ is shown to be positive and equal to ϕ (Martel, 2011a).

1.3.2 Relief of Overburden Hypothesis

The relief of overburden hypothesis posits that the reduction of primary confining pressure that a rock has either solidified under or experienced by burial can act as the fundamental cause for nucleation of sheeting joints (Jahns, 1943). The process involves the unloading of rock weight due to erosion, relieving the compressive stresses on the underlying rock, which should cause the rock to expand. Advocates of this hypothesis equate the radial expansion of the exhumed rock mass with the development of fractures (i.e., sheeting joints) subparallel to the surface (Farmin, 1937; Jahns, 1943).

The granites of New England experienced substantial amounts of erosion, with much of the erosion involving glaciation. This certainly holds true in the Fletcher Granite quarry region, where Jahns (1943) estimates that as much as 34 meters of erosion has occurred locally during Quaternary time (Jahns, 1943). For the purposes of this thesis, the relief of overburden hypothesis will be supported if the highest concentration of sheeting joints is in areas that have had the highest amount of erosion (e.g., valleys and bowls).

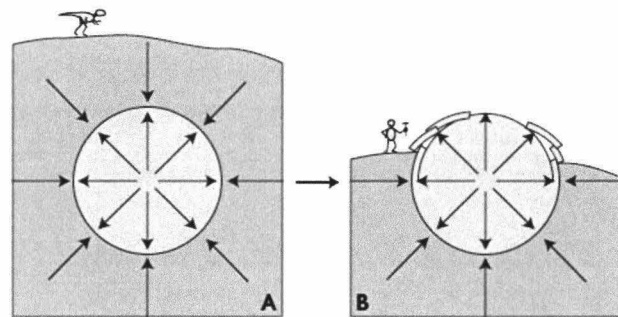


Figure 5. Cartoon depicting the process that creates sheeting joints through relief of overburden. (A) Intrusion solidifies in equilibrium with country rock. (B) Erosion exposes the intrusion causing the rock to exfoliate due to the reduction of confining pressure (modified from Merk, 2010).

1.4 Curvatures

The shape of the topography at Fletcher Granite Quarry must be characterized quantitatively to test hypotheses regarding the formation of sheeting joints. This can be done by evaluating the principal normal curvatures of the topographic surface (Gauss, 1827); these curvatures describe the local shape of a surface at any point on the surface. For a three dimensional surface, two principal curvatures are needed: k_1 and k_2 . The infinitesimal arcs used to obtain the two principal curvatures lie on the surface, where it is intersected by two perpendicular planes; the normal to the surface is a line created by the intersection of

the two planes with each other. If a principal curvature value is positive, the surface is concave along the corresponding arc; if negative, the surface is convex. Different positive and negative values for curvatures k_1 and k_2 define different topographic shapes (Table. 1). Locally, differences in the shapes of the surface generate different combinations of principal curvatures. However, k_1 will always denote the maximum principal curvature, while k_2 will always denote the minimum curvature of the surface. In idealized cases k_1 could be equal to k_2 , as with spheres and planes. In the case of a plane, both principal curvatures are equal to zero. For natural topographic surfaces, principal curvatures invariably are not zero, but can be considered to be zero if the absolute value of the curvature falls below a set threshold value. Quantifying topography in this way supplies a measurable parameter that helps in determining the most likely mechanism for the formation of sheeting joints.

1.5 Calculating Curvature

Curvature is a quantitative way to describe the amount of deviation from a straight line. For a plane curve the curvature vector (K) is defined as the rate of change of the unit tangent (t) of the unit normal of the curve with respect to the change in distance (S) along the curve (Martel, 2011b):

$$K = \frac{dt}{dS} \quad (3)$$

To evaluate the curvature for a three dimensional surface, such as topography, a very similar approach is taken. To find the principal curvature magnitudes and directions, an orthogonal reference frame is set up so the z-axis is aligned with the unit normal of the tangent plane (Fig. 3), which contains both the x- and y-axes.

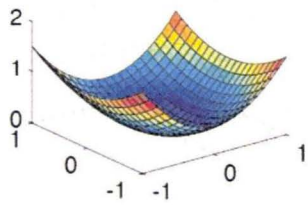
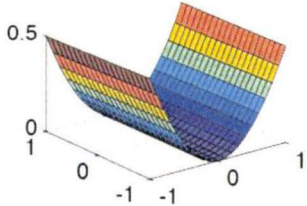
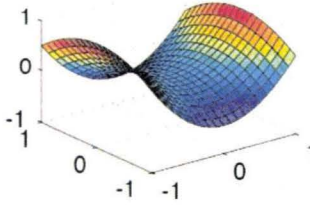
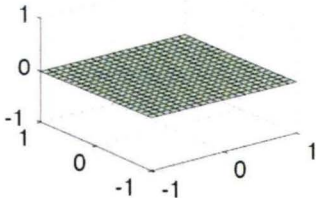
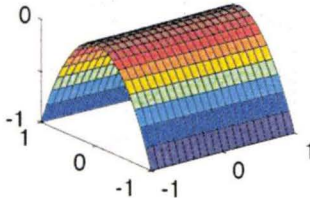
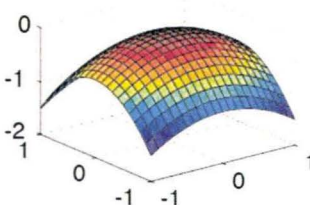
	$k_2 > 0$	$k_2 = 0$	$k_2 < 0$
$k_1 > 0$	Bowl 	Valley 	Saddle 
$k_1 = 0$	k_1 is always $\geq k_2$	Plane 	Ridge 
$k_1 < 0$	k_1 is always $\geq k_2$	k_1 is always $\geq k_2$	Dome 

Table 1. Topographic shapes formed by differing combinations of the principal curvatures k_1 and k_2 . The convex shapes formed by saddles, ridges, and domes are the most likely to develop sheeting joints according to the hypothesis of Martel (2011a). Adapted from Mitchell (2010); images from Martel (2011b).

The first partial derivatives for plane curves in the x-z and y-z planes are the slopes of the curves formed by the intersections of the x-z and y-x planes with the surface. Evaluated at the local coordinate origin, these derivatives are both equal to zero. A Hessian matrix (H) is composed of second order partial derivatives of the surface $Z = Z(x,y)$ in terms of the local elevation (z) and the x and y directions. (Martel, 2011b):

$$[H] = \begin{bmatrix} \frac{\partial^2 z}{\partial x^2} & \frac{\partial^2 z}{\partial x \partial y} \\ \frac{\partial^2 z}{\partial y \partial x} & \frac{\partial^2 z}{\partial y^2} \end{bmatrix} \quad (4)$$

This matrix is symmetric (i.e., the off-diagonal terms are equal). The principal values of the Hessian matrix are the greatest and least normal curvatures k_1 and k_2 . Using matrix multiplication, the Hessian matrix (H) and the infinitesimal vector $[X]$ together yield the slopes of the arcs in the x-z and y-z planes:

$$[H][X] = \begin{bmatrix} \frac{\partial^2 z}{\partial x^2} & \frac{\partial^2 z}{\partial x \partial y} \\ \frac{\partial^2 z}{\partial y \partial x} & \frac{\partial^2 z}{\partial y^2} \end{bmatrix} \begin{bmatrix} dx \\ dy \end{bmatrix} = \begin{bmatrix} \frac{\partial z}{\partial x} \\ \frac{\partial z}{\partial y} \end{bmatrix} \quad (5)$$

The principal curvatures, given by the matrix $[K]$, and the principle directions, given by the vector $[X]$ satisfy the following equation of the surface:

$$[H][X] = k[X] \text{ where } k \rightarrow \begin{bmatrix} k_1 & 0 \\ 0 & k_2 \end{bmatrix} \quad (6)$$

Principal curvatures of a surface are the eigenvalues of the Hessian matrix. The principle directions give the directions in which the surface slope changes most positively and least positively. The above process is straightforward where the local reference frame describing the surface is aligned with the global reference frame. This is always the case at high or low points in the topography (e.g., top of a summit, bottom of a depression) if the global x- and y-axes lie on a horizontal plane. When these two reference frames are not in alignment, which is the case at intermediate elevations, determining the principal curvatures requires a more sophisticated approach (e.g., Struik, 1961; Martel, 2011b).

Chapter 2. Methods

The Fletcher Granite Quarry in Massachusetts was selected for this thesis for several reasons: First, well documented sheeting joints are present and a detailed topographic map (Fig. 6) plots 50 of the locations where they were found (Jahns, 1943; Holzhausen, 1989). Second, the topography is diverse. Third, stresses have been measured at the quarry using overcoring strain gauges at depths as great as 60 meters. The most compressive principal stress measured is 31.2 to 55.5 MPa and trends $\sim 47^\circ$; the least compressive principal stress is 16.2 to 27.0 MPa and trends $\sim 137^\circ$ (Holzhausen, 1989). The topographic contour map of Jahns (1943) details the locations of sheeting joints and is the basis for the analysis done in this thesis (Fig. 6). His map was used because the topography at the Fletcher Granite Quarry was substantially altered by mining operations after Jahns completed his work.

In order to utilize the work of Jahns, the hand-drawn topographic map was cleaned (Fig. 7) and contours from this base map were digitized in MATLAB. The elevation data from the digitized contour lines were then interpolated onto a regular rectangular grid. Initial interpolation results using built-in MATLAB interpolation functions caused a few problems, with two standing out. First, the correct size for the grid was somewhat difficult to pin down. Course grids (e.g., 50 x 50) processed quickly, but did not accurately represent the contour data (Fig. 8a). Very fine grids (e.g., 1000 x 1000 and above) were very detailed but substantially increased processing time. The differences between contour maps made from course-gridded data and fine-gridded data were not readily apparent at this stage (Fig. 8), but when the final curvature data was plotted the lack of detail became very noticeable (Fig. 10, Fig. 11). For the results described in this thesis, a grid of 800 by 800 was used as a compromise between processing speed and accuracy (Fig. 8b). Second, the MATLAB function 'griddata', which was originally used for interpolation, created some unwanted results; the interpolated contours were not staying true to the data in some areas. An attempt at remedying this was to try different methods of interpolation within 'griddata' (e.g., 'nearest', 'linear', and 'v4'). The 'v4' method provided the best results; however some areas still did not come close to matching the original digitized contours (Fig. 8a). Another interpolation function was used that utilizes Green's function for splines in tension, 'spline2d', to try for a better fit (Wessel, 1998). This function allows the user to apply tension to the contour lines to improve the fit (much the same way a spline was used in

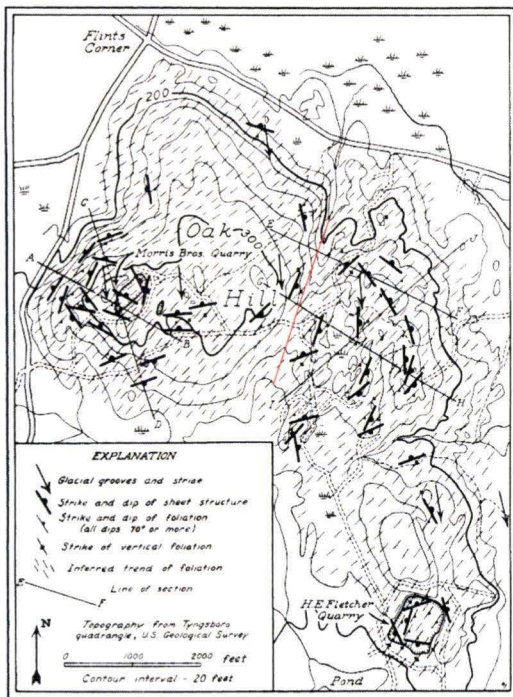


Figure 6. Original topographic map of Fletcher Granite Quarry used as base map for digitizing contours. Contour interval is 20 feet (Jahns, 1943).

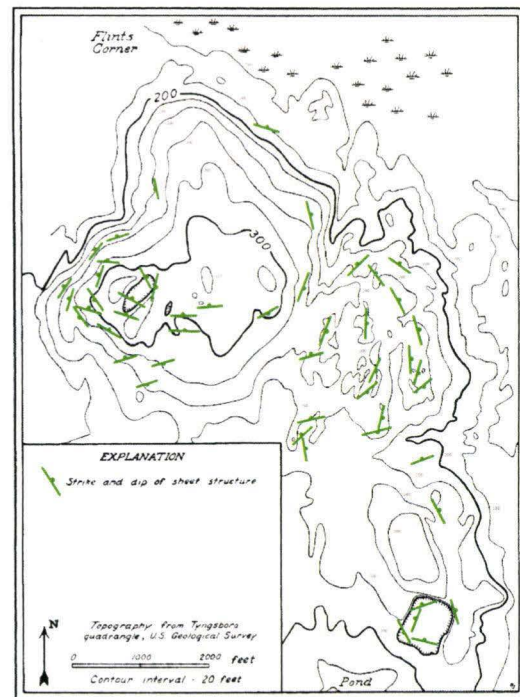


Figure 7. Cleaned version of the Jahns (1943) map. Green symbols show strike and dip direction of sheeting joints. Contour interval is 20 feet.

shipbuilding). However, it did not produce a drastic improvement in accuracy for the problem areas. It was decided a tension value of 0.1 gave satisfactory results as far as the interpolation (Fig. 8b).

The interpolated grid was then filtered to smooth out the topography and remove any small scale features that would cause errors in calculating curvatures; curvatures are calculated using second order derivatives that are very sensitive to small irregularities in the topography. For the final curvature analysis, only features that were large enough to effect the formation of sheeting joints were left unfiltered; these were features with a wavelength of at least 30 meters or so. Filtering was accomplished by setting high- and low-band-pass filters, which are based on radial Gaussian functions centered on low- and high-pass frequency values. Discrete Fourier transforms of the topographic data are multiplied by each filter, and the topographic surface is reconstructed using the inverse of the filtered discrete Fourier transform data set (for more detail, see Perron, 2008). Frequency values are determined based on the inverse of the

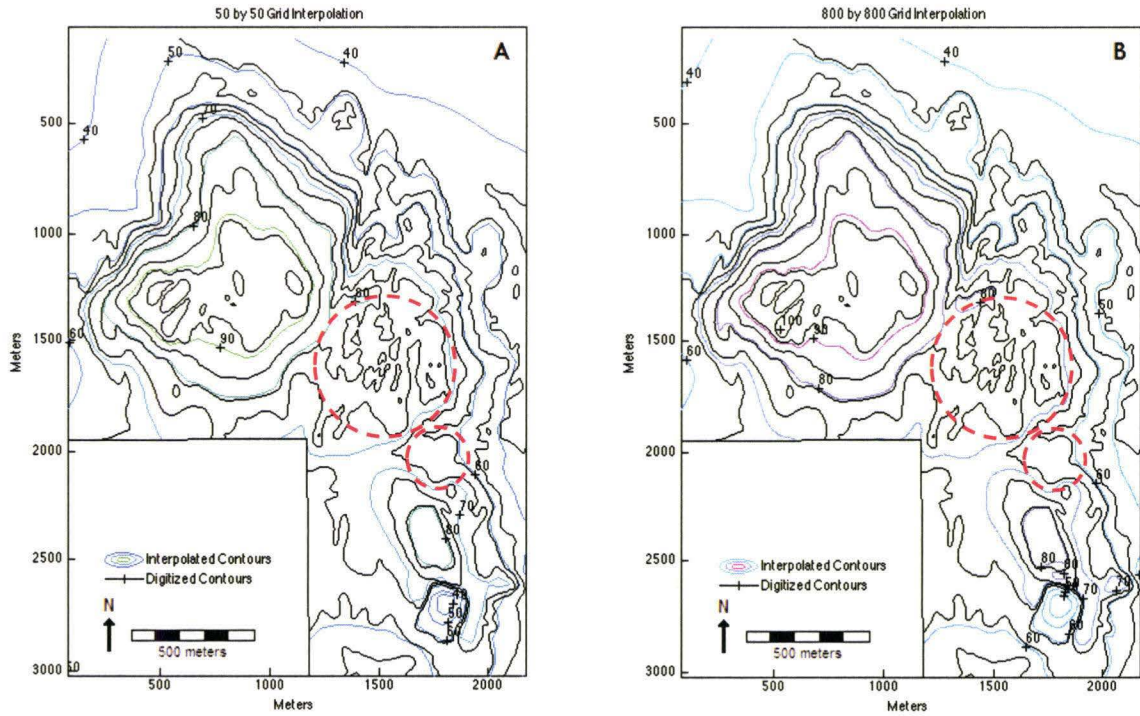


Figure 8. Topographic contour maps prepared from (a) 50 by 50 interpolated grid using the ‘griddata’ function, and (b) 800 by 800 interpolation grid using the ‘spline2d’ function with a tension value of 0.1. Colored contours are interpolated; black contours are digitized from the original Jahns (1943) map. Dashed red circles indicate areas where the interpolated contours do not stay true to the original contours.

wavelength being addressed. This step required quite a bit of trial and error exploration, as the topography of the quarry is rather subdued. The final results shown in this thesis have a high-frequency limit of $3.5 \times 10^{-2} \text{ m}^{-1}$ and a low-frequency limit of $1.0 \times 10^{-3} \text{ m}^{-1}$, with corresponding wavelengths of ~ 30 meters and 1000 meters, respectively.

Principal normal curvatures (k_1 and k_2) of the topography were then evaluated using the codes of Martel (2011). Local topographic forms were classified as one of six features: domes, ridges, saddles, planes, valleys, and bowls (Table 1). These determinations were made based on the magnitudes and sign of k_1 and k_2 ; k_1 being the most positive curvature and k_2 being the least positive curvature. At each point on the topographic surface, the curvatures were compared to values that define each topographic shape. A tolerance level could be set to define the curvature levels that were not rounded off to zero. An overly large (e.g., Fig. 9) tolerance level caused topography to be classified as excessively planar, while an overly small tolerance (e.g., Fig. 10) caused topography to be classified as excessively dome, saddle and

bowl shaped. An acceptable amount of tolerance was determined to be 0.0035 m^{-1} ; principle curvatures with absolute values less than this were set to zero. Several problems were encountered during this stage. First, the interpolation functions yielded unexpected short wavelength artifacts parallel to the long axis of the map (Fig. 9). Their extent raised concerns about the accuracy of the calculated curvatures and the classification of the topographic shapes (Fig. 9). The artifacts seemed to be worst when the 'spline2d' function was used, but were present when using 'griddata' as well. All attempts to remove the vertical artifacts were unsuccessful, and in the end it was only possible to minimize them by adjusting the filter, tension, and tolerance values. This was done mostly by trial and error, as the appearance of the artifacts seemed to be arbitrary. Second, the low quality of coarse grid sizes became very evident, making the topographic shape map pixelated (Fig. 10). This was remedied by interpolating onto a finer grid and rerunning the curvature code. Third, the correct amount of filtering had to be determined. Excessive filtering washed out most of the topographic shapes and filled them in as planar, and in some cases completely distorted the map. This problem was compounded as certain frequencies caused the artifacts

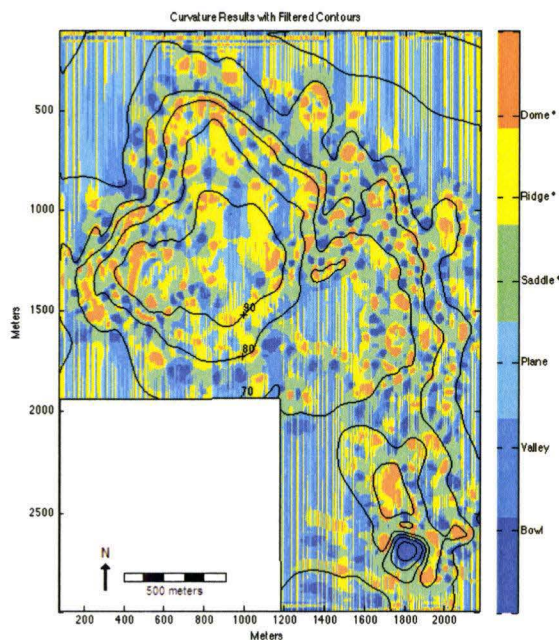


Figure 9. Curvature map illustrating short-wavelength vertical artifacts. Grid size is 800 by 800, interpolated using 'spline2d', a tension of 0.2. The high-frequency limit is $3.5 \times 10^{-2} \text{ m}^{-1}$, and the low-frequency limit is $1.0 \times 10^{-3} \text{ m}^{-1}$.

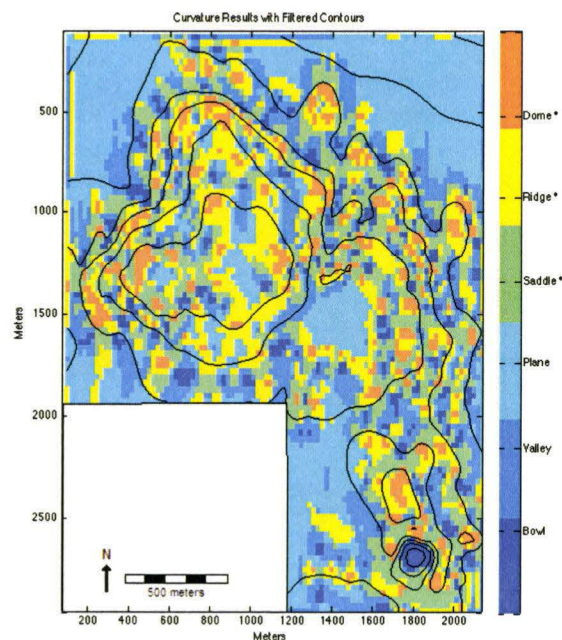


Figure 10. Curvature map illustrating pixilation. Grid size is 50 by 50, interpolated using 'spline2d', a tension of 0.1. The high-frequency limit is $3.5 \times 10^{-2} \text{ m}^{-1}$, and the low-frequency limit is $1.0 \times 10^{-3} \text{ m}^{-1}$.

to improve or worsen. The most robust results were obtained with a combination of filters whose magnitude corresponded closely to the relative wavelength of topographic features present at the Fletcher Granite Quarry. This meant using frequency filters that fit the topography, and were not overly small (e.g., $1 \times 10^{-10} \text{ m}^{-1}$) or overly large (e.g., $1 \times 10^2 \text{ m}^{-1}$), otherwise artifacts worsened. Also keeping the high-pass and low-pass filters within two or three orders of magnitude of each other reduced the presence of artifacts. Slight departures from the final filter values, even an order of magnitude, caused increases in vertical artifacts.

The last step was to overlay the sheeting joint locations that were mapped by Jahns (1943) onto the map produced by curvature analysis to aid in the comparison between the sheeting joints and the local shape of the topographic surface. For each location that had sheeting joints, the topographic shape of the surface was noted and recorded (Table 2). Percentages were then calculated for the sheeting joints located on each topographic shape, of which there were 45 after exclusions. Five sites were excluded due to the ambiguity of the interpolation (Fig. 8) and the small wavelengths of the unfiltered topography there. Excluding these sites improved the reliability of the analysis.

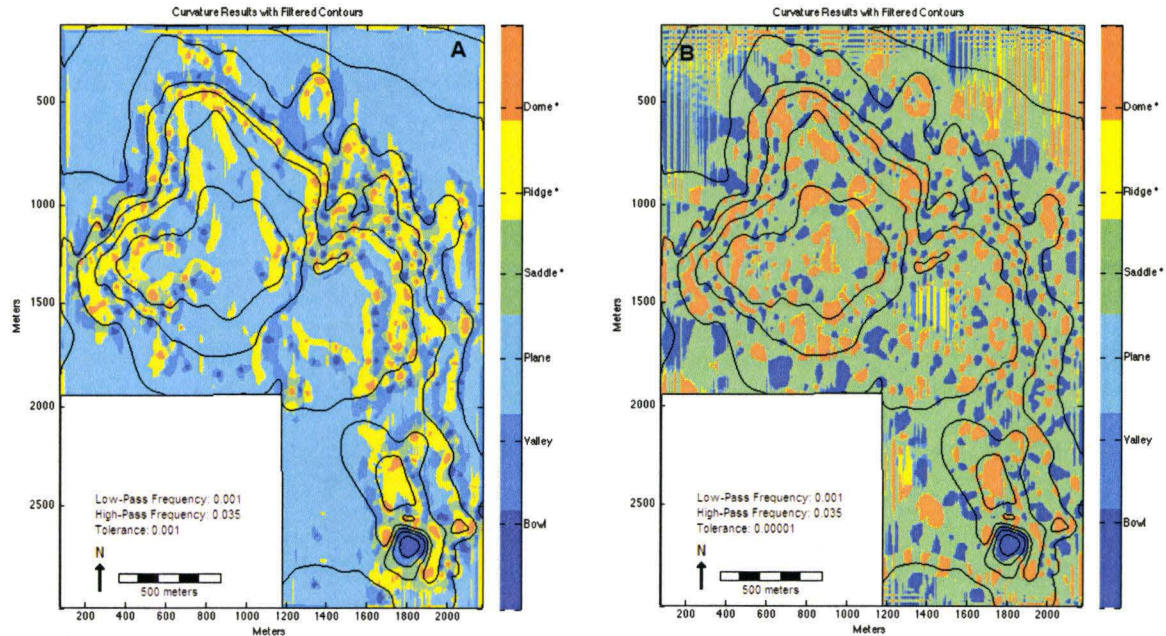


Figure 11. Curvature maps prepared from (a) 800 by 800 interpolated grid using 'spline2d', with a tension of 0.001. The high-frequency limit is $3.5 \times 10^{-2} \text{ m}^{-1}$; the low-frequency limit is $1.0 \times 10^{-3} \text{ m}^{-1}$, and (b) 800 by 800 interpolated grid using 'spline2d', with a tension of 0.00001. The high-frequency limit is $3.5 \times 10^{-2} \text{ m}^{-1}$; the low-frequency limit is $1.0 \times 10^{-3} \text{ m}^{-1}$.

Chapter 3. Results

The results of the principal curvature analysis are shown in Fig. 12 and Fig. 13. A large portion of the map is classified as locally planar (34%), shown in cyan (Fig. 14A). Most of these areas occur along the edges of the map where contour lines on the original map were either lacking or widely spaced. One area classified as planar (red circle, Fig. 8) is made up of small-wavelength, low-amplitude features that created problems during interpolation. These features were filtered out, making the area appear locally planar. Jahns (1943) mapped sheeting joints at five locations within this area, and these were excluded from analysis. An additional point was excluded as it is located in an area that is perforated by vertical artifacts, making the classification of the topographic shape there ambiguous (small dashed circle, Fig. 8). The topographic shapes classified as ridges, saddles, and valleys all have about an equal percentage of occurrences (17%, 19%, and 18%, respectively) and make up the majority of the map. Bowls and domes have an equal percentage of occurrences (6%) and make up the smallest proportion of the map. Of the

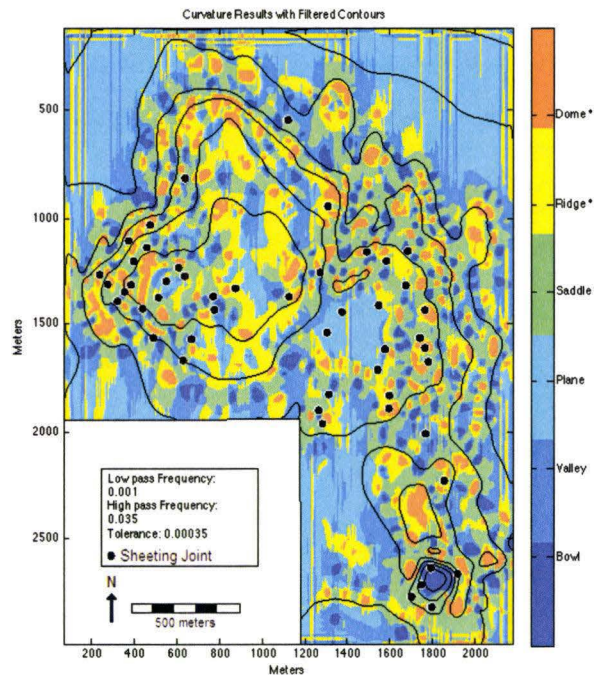


Figure 12. Final curvature map using 'spline2d' for interpolation with a tension of 0.1. Grid size is 800 by 800. The high-frequency limit is $3.5 \times 10^{-2} \text{ m}^{-1}$; the low-frequency limit is $1.0 \times 10^{-3} \text{ m}^{-1}$.

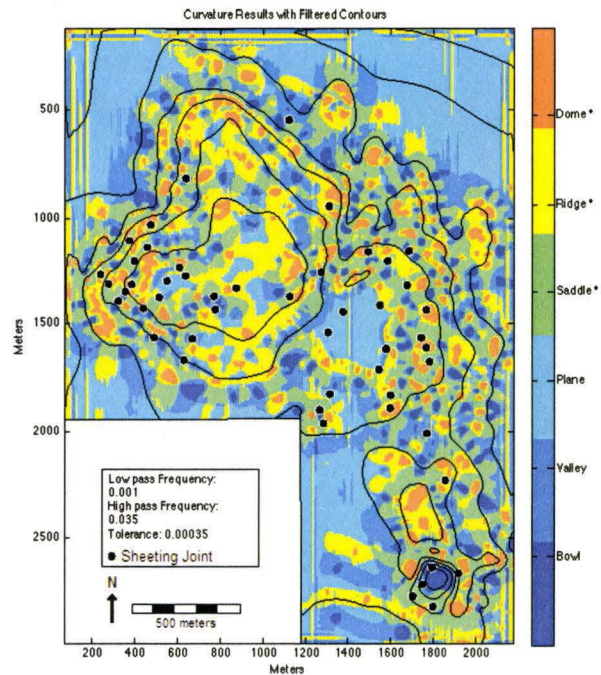


Figure 13. Final curvature map using 'griddata' for interpolation. Grid size is 800 by 800. The high-frequency limit is $3.5 \times 10^{-2} \text{ m}^{-1}$; the low-frequency limit is $1.0 \times 10^{-3} \text{ m}^{-1}$.

Topographic Shape	Dome	Ridge	Saddle	Plane	Valley	Bowl
Percentage of Topographic Shape (%)	6	17	19	34	18	6
# of Sheeting Joints	4	8	18 (19)	1 (5)	8 (9)	5
Percentage of Sheeting Joints (%)	9.1	18.2	40.9	2.3	18.2	11.4

Table 2. Distribution of sheeting joints at Fletcher Granite Quarry, and the topographic shape in which they are located. Numbers in parenthesis indicate values prior to exclusions. Percentages calculated using total values for locations and sheeting joints. Figure 11 was used to tabulate number of sheeting joints for each topographic shape.

remaining 44 locations where sheeting joints were mapped, all but 14 of them are located at domes, ridges, or saddles (Table 2, Fig. 14b). A total of 68.2% of the sheeting joints are located on surfaces that are convex in at least one direction, and only 31.8% of the sheeting joints are located elsewhere. A high percentage (40.9%) of the sheeting joints are located on topography classified as saddle shaped (Fig. 14b). Of the 14 locations where sheeting joints are deemed to be in a valley, a bowl, or a plane, five were within the Morris Brothers and H.E. Fletcher quarries (Fig. 6), and likely formed before the quarries were excavated. In Figure 6 the red line indicates what was thought to be a very broad saddle. Six total sheeting joints were mapped in this area. Of those, two are located in an area classified as a saddle, and two in an area classified as a ridge. The other two points are located in a planar area that is excluded.

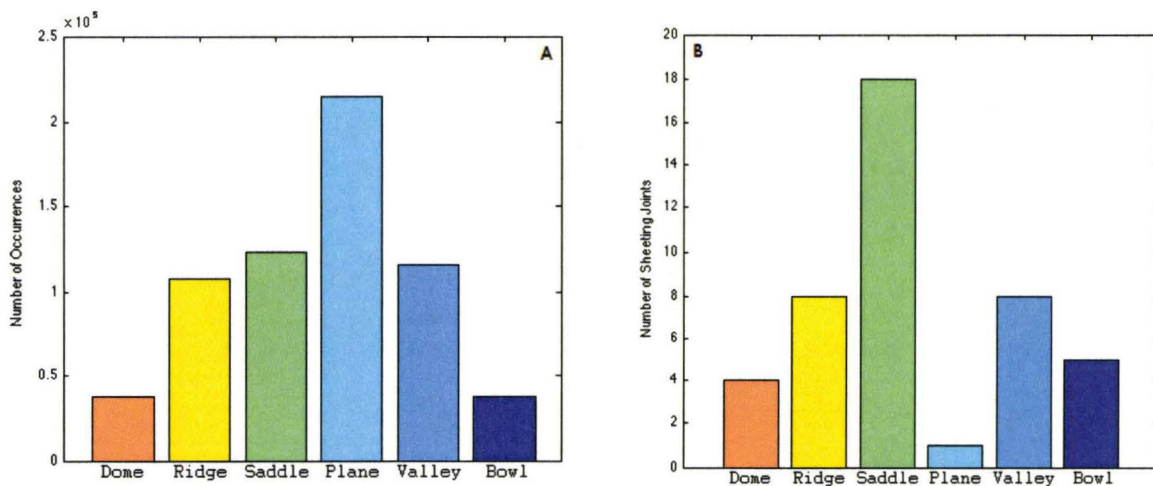


Figure 14. Histograms showing (a) the total number of pixels classified for each topographic shape, and (b) the total number of sheeting joints located on each topographic shape, from Table 2.

Chapter 4. Discussion and Conclusion

This thesis tested two hypotheses that identify a mechanism for the formation of sheeting joints. Curvature of the topographic surface was used as the metric with which to test each hypothesis, as it can favor the rejection or it can support each by determining if sheeting joints are preferentially located on convex or concave surfaces.

4.1 Discussion

The percentage of the map area taken up by each individual topographic shape, as determined from the classification of the principle curvatures, does not seem to reflect the percentage of sheeting joints on that surface. In other words, valleys, ridges and saddles all take up similar percentages of the map, but roughly twice as many sheeting joints are located on topography classified as saddles compared to valleys and ridges (Fig. 14). In fact, almost half of the sheeting joints present at Fletcher Granite Quarry are located on saddle shaped topography. The planar topographic shape accounted for a large portion of the map (43%), but also had the smallest percentage of sheeting joints (2.3%). The percentage of sheeting joints in bowls is somewhat deceptive, as five of the joints located in bowl shaped topography were quarries. If this is taken into account, more sheeting joints are located on dome shaped topography (9.1%) than bowl shaped topography (0%, if the quarries are disregarded)(Table 2).

The presence of the short-wavelength vertical artifacts may have had some effect on the curvature analysis, and in turn this could have affected the classification of topographic shapes. However, the artifacts are concentrated around the edges of the map (which is expected to happen due to the way the curvatures are calculated), and also seem to have bled into areas classified as planar surfaces, most notable in the northern section of the map. Only a few sheeting joints are located in areas where the vertical artifacts noticeably effected the topographic shape classification (most of which are found within the planar area that was excluded for inaccuracies in the interpolation), but all of them have been excluded. Exclusion of these points effectively limits any influence the vertical artifacts may have had over the classification of the topographic shapes, as a specific cause for the vertical artifacts could not be determined within the timeframe of this thesis.

4.2 Conclusions

The overwhelming majority (69%) of sheeting joints mapped by Jahns (1943) is located where the local topographic surface is convex in at least one direction (i.e. domes, ridges, and saddles), while half as many are found elsewhere. This finding supports the hypothesis of Martel (2006, 2011a) that sheeting joints open as a result of tension induced by high surface-parallel compressive stresses acting on topographic surfaces that are convex in at least one direction. Only 14 locations where sheeting joints were mapped are classified as a concave, and only 9 of those are located in a natural bowl or valley. Bowls and valleys are features that logically have been more heavily eroded than their surroundings. If erosion of overburden were the primary mechanism behind the formation of sheeting joints then these areas should contain the highest concentration of sheeting joints, which is not the case. As such, the findings do not favor the mechanism proposed by Jahns (1943) that sheeting joints are primarily a response to the relief of overburden due to erosion.

4.3 Future Research

More research is needed to resolve the issues with short-wavelength vertical noise. Using another program to digitize and interpolate the contours could reduce or completely eliminate the vertical artifacts if the error lies somewhere within MATLAB. This approach was attempted, but processing limitations and bugs with porting the GIS data to MATLAB were encountered. With sufficient resources these issues could be resolved. A larger basemap could reduce artifacts around the borders of the mapping area and possibly prevent artifacts from encroaching into the region of interest. Additional research could be conducted to analyze the stresses in the rock around the Fletcher Quarry using the topographic data obtained from the digitized map and stress measurements from the Fletcher Granite Quarry (Holzhausen, 1979). This analysis could utilize a method based on Fast Fourier Transforms (Perron et al., 2008). Previous research using this approach has only taken into consideration either gravity (Haneberg, 1999) or regional geologic stresses (Holzhausen, 1979), but not both. A more complete analysis would account for both gravity and regional geologic stresses, and give a better picture of the forces involved in creating the sheeting joints found at Fletcher Granite Quarry.

Appendix: Codes

Codes used for digitizing contours

A digital image of the map is required to run this code, the formatting restrictions for this image can be found in the help pages for the 'imread' function in MATLAB. This code was run using MATLAB R2014a Student Edition.

```
function fletcher_digitize_image
% FLETCHER_DIGITIZE_IMAGE - Takes an input file, such as a .tiff,
% and allow points to be chosen from the image using ginput. These points will
% be saved into a .txt file, which can later be assigned to variable to
% plot the x and y values.
% Input: Image file

% Enter data file to be digitized.
A = imread(input('Enter Input File: '));

% Create figure 1 and bring up the image
figure(1);
image(A);
axis equal

% Use ginput to select points on the image to convert them to x and y
% coordinates.
[x,y] = ginput2('r');

% Save the data to a txt file.
A = input('Continue? (1 = Yes, 2 = No): ');
if (A == 1)
    save_contours(x, y);
    % Display the resulting points in figure 2.
    B = input('Load File? (1 = Yes, 2 = No): ');
    if (B == 1);
        load_display;
    end
end

end

%-----

function save_contours(x, y)
% SAVE_CONTOURS - Save the x and y coordinates obtained to a txt file.
% If no file exists, one will be created. If the file name already exists
% the data will be appended to the preexisting txt file.

% Combines x and y coordites into one matrix.
points(:,1) = round(x);
points(:,2) = round(y);

% Asks for file name.
file = input('Enter Save File: ');
```

```

% If statement prevents overwriting of previously obtained values.
if exist(file, 'file') == 2
    save(file, 'points', '-append', '-ascii');
else
    save(file, 'points', '-ascii');
end

%-----

function load_display
    % LOAD_DISPLAY - This function will load a .txt file with x and y
    % coordinates seperated into two columns and plot them.

% Loads file and assigns it to a variable.
P = load(input('Input Load File: '));

% Allows user to choose figure number
f = input('Figure: ');
% Pulls out x and y coordinates for plotting.
x = P(:,1);
y = P(:,2);

% Plot the x and y coordinates.
figure(f);
hold on
axis ij;
plot(x,y);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

Plotting digitized contours

It is necessary to combine all the files containing the digitized contour data in order to plot a full map. The function 'fletcher_joints' is a similar function that loads a .txt file containing the locations of individual points on the digitized map, in this case the locations of sheeting joints. This function can be used to create an overlay to check the accuracy of interpolation and filtering. This code was written and run using MATLAB 2014a Student Edition.

```

function fletcher_complete_map
% FLETCHER_COMPLETE_MAP - Loads the contour data from
% fletcher_digitize_image and plots the data as seperate polygons using
% load_data2.

figure(1)
axis equal
hold on
axis ij

% Load Sheeting Joint Locations

```

```

fletcher_joints;

load_data2('140c1.txt', 'k');
load_data2('140c2.txt', 'k');
load_data2('140c3.txt', 'k');
load_data2('160c.txt', 'k');
load_data2('180c.txt', 'k');
load_data2('180c1.txt', 'k');
load_data2('180c2.txt', 'k');
load_data2('180c3.txt', 'k');
load_data2('180c4.txt', 'k');
load_data2('180Poc1.txt', 'k');
load_data2('180Poc2.txt', 'k');
load_data2('200c1.txt', 'k');
load_data2('200c2.txt', 'k');
load_data2('200Quc.txt', 'k');
load_data2('220c.txt', 'k');
load_data2('220c1.txt', 'k');
load_data2('240c.txt', 'k');
load_data2('240c1.txt', 'k');
load_data2('240c2.txt', 'k');
load_data2('260c.txt', 'k');
load_data2('260c1.txt', 'k');
load_data2('260c2.txt', 'k');
load_data2('260c3.txt', 'k');
load_data2('260c4.txt', 'k');
load_data2('260c5.txt', 'k');
load_data2('260c6.txt', 'k');
load_data2('260c7.txt', 'k');
load_data2('260c8.txt', 'k');
load_data2('260c9.txt', 'k');
load_data2('260c10.txt', 'k');
load_data2('260c11.txt', 'k');
load_data2('260c12.txt', 'k');
load_data2('260c13.txt', 'k');
load_data2('260c14.txt', 'k');
load_data2('280c.txt', 'k');
load_data2('300c.txt', 'k');
load_data2('320c1.txt', 'k');
load_data2('320c2.txt', 'k');
load_data2('320c3.txt', 'k');
load_data2('320c4.txt', 'k');
load_data2('320c5.txt', 'k');
load_data2('320c6.txt', 'k');
load_data2('320c7.txt', 'k');
load_data2('320c8.txt', 'k');
load_data2('Border.txt', 'k');
%-----
function load_data2 (x, s)
    % LOAD_DATA - This function will load a .txt file with x and y
    % coordinates seperated into two columns and plot them. Inputs are file
    % name and figure number.
    % Example load_data2('text.txt', 1)

```



```

hold on
axis ij
axis equal

% Loads file and assigns it to a variable.
P = load(x);

% Pulls out x and y coordinates for plotting.
x1 = P(:,1);
y1 = P(:,2);

% Scales the map to meters
xscale = 0.68499;
yscale = 0.6681;
x2 = x1/xscale;
y2 = y1/yscale;

% Plot the x and y coordinates.
plot(x2,y2, s);
%-----
function fletcher_joints
    % FLETCHER_JOINTS - Plots sheeting joint locations from Jahns 1943 map
    % of Fletcher Granite Quarry.

% Loads sheeting joint data and assigns it to the variable.
joints = load('joints.txt')
% Pulls out x and y coordinates for plotting.
joints_x = joints(:,1);
joints_y = joints(:,2);

% Scales the coordinates to meters
xscale = 0.68499;
yscale = 0.6681;
joints_x2 = joints_x/xscale;
joints_y2 = joints_y/yscale;

% Plots the sheeting joint locations
plot(joints_x2, joints_y2, 'o',...
     'MarkerSize', 8,...
     'MarkerEdgeColor', 'w',...
     'MarkerFaceColor', 'k')
End

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

Codes for interpolating digital contours

Digitized contours can be loaded in and saved using this function. User can choose to interpolate using 'griddata' or 'spline2d'. The code for 'spline2d' was obtained from Wessel (1998). Code was written and run using MATLAB 2014a Student Edition.

```

function [Xg, Yg, Zg, xval, yval, zval] = fletcher_interpolate_v4(N,t)
% FLETCHER_INTERPOLATE - Loads in contour data taken using
% fletcher_digitize_image, scales the data into meters, and interpolates it
% onto a grid. A grid with the specified dimensions is created and passed
% through spline2d (or griddata) with the contour data to be interpolated.
% The grid is an N by N matrix of linear equally spaced points. The grid
% can be any size, however a very fine grid will substantially increase
% processing time, while a very course grid will not accurately represent
% the contour data. Contour data is then interpolated over the grid with
% the specified amount of tension. Tension values are between 0 and 1, with
% zero having no tension and 1 the maximum amount of tension. Tension
% values closer to 1 can cause distortion in the data, and as such do not
% accurately represent the data. The fit is then determined by
% fletcher_best_fit, which returns a value, Z. This value varies, but
% generally a lower value indicates a better mathematical fit % to the
% initial contour data. Values for x, y, and z are then saved to a file
% specified by the user.
%
% INPUT:
% N = grid size
% t = tension
%
% OUTPUT:
% Xg = interpolated X data
% Yg = interpolated Y data
% Zg = interpolated Z data
%
%
% EXAMPLE:
% [Xg, Yg, Zg, Z] = interpolate_fletcher(200,.6)
%
% FUNCTIONS USED:
% spline2d
% griddata
%
% VERSION: 4 (March, 5 2015)
% MATLAB: 8.3.0.532 (R2014a)
% AUTHOR: Zachary T. Olds

%-----

% Load the data
[xval1, yval1, zval1] = load_data;

% Scale x and y values to meters
[xval, yval] = thesis_scale_meters(xval1,yval1);

% % Scale z values to meters
zval = zval1 *.3048;

% Grid
rangeX = linspace(min(xval),max(xval), N);
rangeY = linspace(min(yval),max(yval), N);
[Xg, Yg] = meshgrid(rangeX, rangeY);

```

```

% Interpolate via griddata
% Zg = spline2d(Xg, Yg, xval, yval, zval, t);
Zg = griddata(xval,yval,zval,Xg,Yg, 'v4');

% Graph Data
graph_c(Xg, Yg, Zg);
% mesh(Xg, Yg, Zg);
% plot3(xval,yval,zval,'o')

% Save Contour Data
save_c(Xg, Yg, Zg);

%-----

function [xval, yval, zval] = load_data
% Loads contour data from .txt files to xval, yval, and zval variables.

% Load the Data
% 140 Contour
c140 = [load('140c1.txt');
load('140c2.txt');
load('140c3.txt')];
x140 = c140(:,1);
y140 = c140(:,2);
[m,n] = size(c140);
z140 = 140*ones(m,1);

% 160 Contour
c160 = load('160c.txt');
x160 = c160(:,1);
y160 = c160(:,2);
[m,n] = size(c160);
z160 = 160*ones(m,1);

% 180 Contour
c180 = [load('180c.txt');
load('180c1.txt');
load('180c2.txt');
load('180c3.txt');
load('180c4.txt');
load('180Poc1.txt');
load('180Poc2.txt')];
x180 = c180(:,1);
y180 = c180(:,2);
[m,n] = size(c180);
z180 = 180*ones(m,1);

% 200 Contour
c200 = [load('200c1.txt');
load('200c2.txt');
load('200Quc.txt')];
x200 = c200(:,1);
y200 = c200(:,2);

```

```

[m,n] = size(c200);
z200 = 200*ones(m,1);

% 220 Contour
c220 = [load('220c.txt');
load('220c1.txt')];
x220 = c220(:,1);
y220 = c220(:,2);
[m,n] = size(c220);
z220 = 220*ones(m,1);

% 240 Contour
c240 = [load('240c.txt');
load('240c1.txt');
load('240c2.txt')];
x240 = c240(:,1);
y240 = c240(:,2);
[m,n] = size(c240);
z240 = 240*ones(m,1);

% 260 Contour
c260 = [load('260c.txt');
load('260c1.txt');
load('260c2.txt');
load('260c3f.txt');
load('260c4.txt');
load('260c5.txt');
load('260c6.txt');
load('260c7.txt');
load('260c8.txt');
load('260c9.txt');
load('260c10.txt');
load('260c11.txt');
load('260c12.txt');
load('260c13.txt');
load('260c14.txt')];
x260 = c260(:,1);
y260 = c260(:,2);
[m,n] = size(c260);
z260 = 260*ones(m,1);

% 280 Contour
c280 = load('280c.txt');
x280 = c280(:,1);
y280 = c280(:,2);
[m,n] = size(c280);
z280 = 280*ones(m,1);

% 300 Contour
c300 = load('300c.txt');
x300 = c300(:,1);
y300 = c300(:,2);
[m,n] = size(c300);
z300 = 300*ones(m,1);

```

```

% 320 Contour
c320 = [load('320c1.txt');
load('320c2.txt');
load('320c3.txt');
load('320c4.txt');
load('320c5.txt');
load('320c6.txt');
load('320c7.txt');
load('320c8.txt')];
x320 = c320(:,1);
y320 = c320(:,2);
[m,n] = size(c320);
z320 = 320*ones(m,1);

% Concatenate X Data Values
xval = [x140; x160; x180; x200; x220; x240; x260; x280; x300; x320];
% Concatenate Y Data Values
yval = [y140; y160; y180; y200; y220; y240; y260; y280; y300; y320];
% Concatenate Z Data Values
zval = [z140; z160; z180; z200; z220; z240; z260; z280; z300; z320];

%-----

function [ xscaled, yscaled ] = thesis_scale_meters( xvalue, yvalue )
% THESIS_SCALE - Scales values collected by ginput and converts them to the
% map scale of Jahns 1943, in units of miles.

% Scale factor in feet
xfeet = 7150.9433;
yfeet = 9924.5283;

% Convert to Meters
xmeters = xfeet * .3048;
ymeters = yfeet * .3048;

% Data x and y axis
xdata = max(xvalue);
ydata = max(yvalue);
% disp(['xdata = ', num2str(xdata)])
% disp(['ydata = ', num2str(ydata)])

% Meters
xscale = xdata/xmeters;
yscale = ydata/ymeters;
% disp(['x scale value = ', num2str(xscale)]);
% disp(['y scale value = ', num2str(yscale)]);

% Scale x and y values
xscaled = xvalue / xscale;
yscaled = yvalue / yscale;

%-----

```

```

function graph_c(Xg, Yg, Zg)
% Makes a contour plot of the data in meters, with contour labels.
% Example: graph_c(Xg, Yg, Zg)

figure(1)
hold on
axis equal; axis ij;

% % Plot in feet
% int = 1400:200:3200;
% con = contour(Xg,Yg,Zg, int);

% Plot in meters
int = 40:10:100;
con = contour(Xg,Yg,Zg, int);
xlabel('Meters');
ylabel('Meters');
clabel(con);

% Cover contours in NaN area
% xnan = [.04626, .7275, .7321, .04626];
% ynan = [1.211, 1.207, 1.881, 1.881];
% fill(xnan,ynan, 'w');
%load_data3('Border.txt');
hold off

%-----

function save_c(Xg,Yg,Zg)
% Saves x, y, and z interpolated contour data to three separate text files.
% Example: save_c(Xg, Yg, Zg);

% Saves x values
A = input('Save Data? (1 = Yes, 2 = No): ');
if (A == 1)
    file = input('Enter Save File for X Value: ');
    save(file, 'Xg', '-ascii');

    file = input('Enter Save File for Y Value: ');
    save(file, 'Yg', '-ascii');

    file = input('Enter Save File for Z Value: ');
    save(file, 'Zg', '-ascii');
end

%%%%%%%%%%

```

Curvature codes for calculating principal curvatures

Code was obtained from Martel (2011a) and is used by 'fletcher_curvature_v3' in order to obtain curvatures from a regular grid. For the full code see the supplement to Martel (2011a). Code was run using MATLAB 2014a Student Edition.

```

function [X_,Y_,Z_,K,H,k1,k2,a1,b1,g1,a2,b2,g2,dzx,dzy] = curvcode4(X,Y,Z)
% function [X_,Y_,Z_,K,H,k1,k2,a1,b1,g1,a2,b2,g2,dzx,dzy] = curvcode4(X,Y,Z)
% Calculates Gaussian (K) and mean curvature (H), most positive and
% least positive principal curvatures (k1 and k2, respectively), and
% the associated direction cosines for k1 and k2 (a1,b1,g1, and
% a2, b2, and g2, respectively) for a surface based on regular grids.
% This code uses the shape operator approach of Smith and Sequin.
% http://www.cs.berkeley.edu/%7Esequin/CS284/TEXT/diffgeom.pdf
%
% The following functions are required to run this code:
% smthseq      Main function to calculate curvatures
% eig2d        Returns eigenvalues and eigenvectors for 2D matrices
%              This is an alternative to MATLAB's function eig
% These functions should be in the same directory (folder) as curvcode4
%
% Input parameters
% X = grid of eastings (east coordinates),
% Y = grid of northings (north coordinates),
% Z = grid of elevations.
% X,Y, and Z should have the same dimensions
%
% Input parameters
% X = regular grid of eastings (east coordinates)
% Y = regular grid of northings (north coordinates)
% Z = regular grid of elevations
% Note: X,Y, and Z should have the same dimensions
%
% Output parameters
% X_ = regular grid of eastings (east coordinates) = X(2:m-1,2,n-1);
% Y_ = regular grid of northings (north coordinates) = Y(2:m-1,2,n-1);
% Z_ = regular grid of elevations = Z(2:m-1,2,n-1);
% K = Gaussian curvature (k1*k2)
% H = Mean curvature (H) (k1 + k2)/2
% k1 = most positive principal curvature
% k2 = most negative principal curvature
% a1,b1,g1 = direction cosines for k1 with respect to the x,y,z axes
% a2,b2,g2 = direction cosines for k2 with respect to the x,y,z axes
% % See online notes for GG303 for more on direction cosines

% Examples
% [X,Y] = meshgrid(-2:0.1:2,-1:0.1:1);
% % Test 1: Parabolic cylinder parallel to x-axis
% Z = X.^2;
% [X_,Y_,Z_,K,H,k1,k2,a1,b1,g1,a2,b2,g2,dzx,dzy] = curvcode4(X,Y,Z)
% % Test 2: Parabolic cylinder parallel to y-axis
% Z = Y.^2;
% [X_,Y_,Z_,K,H,k1,k2,a1,b1,g1,a2,b2,g2,dzx,dzy] = curvcode4(X,Y,Z)
% % Test 3: Parabolic cylinder parallel with axis in xy plane
% t = pi/4*ones(size(X));
% cost = cos(t); sint = sin(t);
% X = (X.*cost + Y.*sint).^2;
% [X_,Y_,Z_,K,H,k1,k2,a1,b1,g1,a2,b2,g2,dzx,dzy] = curvcode4(X,Y,Z)
% % Test 4: Monkey saddle z = x.^3 - 3xy.^2;
% Z = X.^3 - 3*x.*y.^2;

```

```

% [X_,Y_,Z_,K,H,k1,k2,a1,b1,g1,a2,b2,g2,dzx,dzy] = curvcode4(X,Y,Z)
% Test 5: Egg carton z = cos(x).*cos(y);
% Z = cos(X).*cos(Y);
% [X_,Y_,Z_,K,H,k1,k2,a1,b1,g1,a2,b2,g2,dzx,dzy] = curvcode4(X,Y,Z)

% Calculate grid spacing
dx = X(1,2) - X(1,1);
% Cautionary note!! In Matlab, row numbers increase from
% the top of a page to the bottom, rather than the bottom to the top.
% This can cause derivatives AND GRADIENTS in the y-direction to have an
% unexpected sign or to appear flipped about the x-axis (horizontal axis)
% if not handled with the Matlab convention in mind.
dy = (Y(2,1) - Y(1,1));

[m,n] = size(X);
% Trim edge rows and columns from input matrices because good values
% for second derivatives cannot be obtained there.
X_ = X(2:m-1,2:n-1);
Y_ = Y(2:m-1,2:n-1);
Z_ = Z(2:m-1,2:n-1);

% Find first and second partial derivatives of Z with respect to X and Y.
[dzx,dzy,dzxx,dzyy,dzxy] = derivatives (Z,dx,dy);

% Calculate curvatures and direction cosines for principal directions
[K,H,k1,k2,a1,b1,g1,a2,b2,g2] = smthseq(dzx,dzy,dzxx,dzyy,dzxy);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [dzx,dzy,dzxx,dzyy,dzxy] = derivatives (z,dx,dy)
% function [dzx,dzy,dzxx,Dzyy,dzxy] = derivatives (z,dx,dy)
% Calculates first and second derivatives of function z with respect to x
% and y. The function uses a finite difference method on a 9-spot
% (tic-tac-toe) pattern. The matrices of derivatives returned lack values
% for the first and last rows of z and the first and last columns of z.
[m,n] = size(z);
dzx = zeros(m-2,n-2);
dzy = dzx;
dzxx = dzx;
dzyy = dzx;
dzxy = dzx;

z11 = z(1:m-2,1:n-2);  z12 = z(1:m-2,2:n-1);  z13 = z(1:m-2,3:n);
z21 = z(2:m-1,1:n-2);  z22 = z(2:m-1,2:n-1);  z23 = z(2:m-1,3:n);
z31 = z(3:m ,1:n-2);  z32 = z(3:m ,2:n-1);  z33 = z(3:m ,3:n);

dzx = ( z23 - z21 )/(2*dx);    % Central difference method
dzy = ( z12 - z32 )/(2*dy);    % Central difference method
dzxx = ( z23 - 2*z22 + z21 )/(dx.^2);
dzyy = ( z12 - 2*z22 + z32 )/(dy.^2);
dzxy = ( z13 - z11 - z33 + z31 )/(4*dx*dy);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```


Code to filter topography

This code was obtained from Martel (2011a). It is used by 'fletcher_curvature_v3' to filter topography.

This prevents curvature results from returning bad values due to high frequency, small wavelength features.

```
function [X,Y,Zhp,Zlp] = fletcher_topo_filter_short_v2 (X,Y,Z,flo,fhi)
% function [X,Y,Zhp,Zlp] = topo_filter_short (X,Y,Z,flo,fhi)
% Code to filter topographic data
% Modified from codes example.m and mima.m of Perron, which are
% sample scripts demonstrating use of DEM spectral analysis routines in
% Perron, J.T. et al. (2008), Spectral signatures of characteristic spatial
% scales and non-fractal structure in landscapes, J. Geophys. Res.
% Figure and equation numbers refer to those in the JGR paper.
% I have tended to adopt the notation of mima.m, which is a bit different
% from that of example.m.
% NOTE!!: I have stripped out the de-noising here from topo_filter
% NOTE!!: Perron DOES NOT window the data in his de-noising routine
%
% GENERAL ORGANIZATION OF CODE
% 00. Settings
% 0. View data
% 1. Detrend data
% 2. Prepare to pad and window data
% 3. Transform data from spatial domain to spectral domain
% 4. Plot spectra
% 5. Design filter(s)
% 6. Filter data (This is here example.m leaves off)
% 7. Transform filtered data back to spatial domain (mima.m does this)
% 8. De-window returned data (Perron's codes does not do this far)
% 9. Retrend data
% 10. Save the filtered, re-trended data

%
% Output parameters
% Xt,Yt = matrices of eastings and northings, trimmed to even dimensions
% Zhp = filtered matrix of high-pass elevations (high frequencies passed)
% Zlp = filtered matrix of low-pass elevations (low frequencies passed)
%
% Input parameters
% X,Y = matrices of eastings and northings, respectively
% Z = matrix of elevations (this is what will be filtered)
% flo = frequency at which the high-pass filter starts to increase
%       appreciably above zero
% fhi = frequency at which the high-pass filter reaches 1.
%
% What about the frequency limits? Wavelength is equal to the inverse of
% frequency. The shortest wavelength that can be resolved by 2D data is
% oriented diagonally, and is equal to  $\sqrt{dx^2 + dy^2}$ , or  $\sqrt{2}$  for
% our data (Fig. 4). The longest wavelength that can be resolved is
% infinite in principle, but in practice we wouldn't feel comfortable
% trying to detect any signal with a wavelength longer than our dataset. To
% be even more conservative, the longest wavelength we trust should be a
```

```

% few times shorter than the width of the DEM.
%
% Example
% clear
% Double click on desired files
%
% Example 1
% X=XTC_wide; Y = YTC_wide; Z = ZTC_wide; topo_filter;
% flo = 1e-3; fhi = 5e-3;
% [Xtf,Ytf,Zhp,Zlp] = topo_filter_short(XTC_wide,YTC_wide,ZTC_wide,flo,fhi);
% topo_shapes(Xtf,Ytf,Zlp,0.001)
% Example 2
% flo = 1e-3; fhi = 5e-3;
% [Xtf,Ytf,Zhp,Zlp] = topo_filter_short(XOB_wide,YOB_wide,ZOB_wide,flo,fhi);
% topo_shapes(Xtf,Ytf,Zlp,0.001)
% Example 3 - This looks pretty good for Tenaya Canyon
% flo = 5e-3; fhi = 0.0125;
% [Xtf,Ytf,Zhp,Zlp] = topo_filter_short(XTC_wide,YTC_wide,ZTC_wide,flo,fhi);
% topo_shapes(Xtf,Ytf,Zlp,0.001)
% Example 4 - This looks pretty good for Olmsted Bowl
% flo = 5e-3; fhi = 1/100;
% [Xtf,Ytf,Zhp,Zlp] = topo_filter_short(XOB_wide,YOB_wide,ZOB_wide,flo,fhi);
% topo_shapes(Xtf,Ytf,Zlp,0.001)

% 00. SETTINGS %

%% A note on cell mode

% This script uses Matlab's cell mode, which is why the section where your
% cursor currently resides is highlighted. The easiest way to progress
% through this tutorial is with the following commands, which are also
% available under the "Cell" menu:
%
% Cmd + Enter: Evaluate all the commands in the current cell
% or
% Shift + Cmd + Enter: Evaluate current cell and advance to the next cell
%
% (On a PC, replace Cmd with Ctrl)

%% Set up the Matlab desktop

% First, take a moment to dock the Editor so you can see this script and
% the command line at the same time.

% Make new figures docked and white
%set(0,'DefaultFigureWindowStyle','docked','DefaultFigureColor','w')

% 0. View data %

% Preliminary matters: point spacing on grid and grid dimensions
x = X(1,:);
y = Y(:,1);
dx = abs(x(2) - x(1)); dy = abs(y(2)-y(1)); % grid spacing.
[Ny Nx] = size(Z);

```

```

% Taylor Perron's codes won't run unless both dimensions of Z are even.
% A row or column needs to be trimmed if Nx or Ny is odd.
% This is different from padding a matrix so its dimensions are a POWER of
% two to optimize the performance of an FFT.
if rem(Nx,2) == 1;
    X = X(:,1:Nx-1);
    Y = Y(:,1:Nx-1);
    Z = Z(:,1:Nx-1);
end
if rem(Ny,2) == 1;
    X = X(1:Ny-1,:);
    Y = Y(1:Ny-1,:);
    Z = Z(1:Ny-1,:);
end
%%
x = X(1,:);
y = Y(:,1);
[Ny Nx] = size(Z);

% Display a shaded relief map % COULD ALSO USE HILLSHADE2
% figure('Name','Fig. 2: Shaded relief','NumberTitle','off')
figure(1)
hillshade(Z,x,y,'azimuth',45,'altitude',45,'plotit');
hold on
title('Shaded Relief Map');
xlabel('Meters');
ylabel('Meters');
axis equal
axis ij
hold off

% 1. Detrend data %

% Remove the first-order (planar) trend from the data
% This is to make data stationary in a least-squares sense
% This is important because spectral methods assume data are stationary

Zo = Z; % Save the original elevations for later
[D,P] = detrend4(Z); % D is detrended data, P is best-fit plane

% 2. Prepare to pad and window data %

% Padding: the fast Fourier transform proceeds fastest if the dimensions of
% the input matrix are integer powers of two, which we can achieve by
% padding the DEM with zeros.
% Windowing: the edges of our DEM are not perfectly periodic, the
% spectrum can become contaminated by frequencies used to "fit" the edge
% discontinuity. This occurs in spite of the de-trending.
% One can mitigate this effect by multiplying the DEM by a
% function that tapers to zero at the edges.
pad = 1; % 1 means pad the data with zeros to a power of 2, 0 no padding
window = 1; % 1 means window the data prior to taking the FFT, 0 no window
% 3. Calculate 2D FFT and power spectra for detrended DEM (D)%

```

```

% Note that the zero padding and windowing happen inside fft2D.

[Pm fm Pv fv] = fftdem(D,dx,dy,pad>window);

% A few words about the output from this step:
%
% The Discrete Fourier Transform (DFT) periodogram is calculated here. This
% is different from the power spectral density, which is also commonly used
% as an estimate of the power spectrum.
%
% Pm is the 2D power spectrum matrix, and fm is the corresponding matrix of
% radial frequencies. Pv is a vector version of the power spectrum, and fv
% is the corresponding vector of frequencies. Units of Pm and Pv are
% [units of Z]^2, which in our case is length^2 since Z is a matrix of
% elevations. Units of fm and fv are [units of x]^-1, which for us is
% 1/length since x is distance. Fig. 3 shows an example of a simple surface
% and its 2D power spectrum.
%
% What do the units mean? The periodogram is a measure of how much of the
% original elevation field's variance falls within a given frequency range.
% You can check that the sum of the periodogram is roughly equal to the
% variance in Z. (It will be somewhat less due to the zero padding.)
%
% 4. Calculate 2D FFT and power spectra %
% Note that the zero padding and windowing happen inside fft2D.

%% 4a. 1D Power Spectrum

% Plot the 1D version of the spectrum (Fig. 5). We'll plot the 2D spectrum
% a few steps later, but for now it's easier to visualize in 1D.
% s1d = figure('Name','Fig. 5: 1D Spectrum','NumberTitle','off');
figure(2)
subplot(2,1,1)
[axf axw] = SpecPlot1D(fv,Pv);

% Do we see anything at a 10m wavelength? Yes, there is a broad peak. The
% shorter-wavelength peaks are just pixel-scale noise.
hold(axw,'on');
plot([10 10],get(axw,'ylim'),'k');

%% 4b. Background Spectrum

% Now let's see what the spectrum looks like in 2 dimensions. If we just
% looked at it as is, we wouldn't see much: as the 1D spectrum shows,
% longer-wavelength signals have much higher power, and would swamp any
% shorter-wavelength detail. To make the 2D spectrum easier to view, we'll
% remove the power-law background trend. Because there are so many more
% points at higher frequencies, we'll bin the 1D spectrum and fit the trend
% to the binned values (Fig. 5).

nbin = 20; % Number of bins
B = bin(log10(fv),log10(Pv),nbin,0); % Bin the log-transformed data

% Plot the binned values

```

```

hold(axf,'on');
plot(axf,10.^B(:,1),10.^B(:,2),'ok','markerfacecolor','w');

% Fit a trend with the form  $P \sim 1/f^n$ , and plot it
fit = robustfit(B(:,1),B(:,2));
plot(axf,10.^B(:,1),10.^fit(1)*(10.^B(:,1)).^fit(2),'k');

%% 4c. 2D Power Spectrum

% Use this fit to normalize the 2D spectrum
Pmn = Pm./(10.^fit(1)*fm.^fit(2));

% Plot the normalized 2D spectrum (Fig. 6)
%figure('Name','Fig. 6: 2D Spectrum','NumberTitle','off')
figure(3)
SpecPlot2D(fm,log10(Pmn));

% Clean up workspace
clear Pm fm Pv nbin B axf axw fit Pmn

% 5. Filtering %

%% High-pass filter

% Now let's isolate the signal associated with the features of interest.
% The 1D and 2D spectra suggest that the peak associated with the mounds
% begins at frequencies of  $\sim 0.03$  to  $0.05 \text{ m}^{-1}$  (wavelengths of 20 to 30m).
% The mounds are only a few pixels wide, so we want to retain all
% frequencies higher than about 0.04. We can accomplish this with a
% high-pass filter.

% Plot the filter alongside the 1D spectrum to illustrate its shape (Fig.
% 5). Note how the filter transitions smoothly from 0 to 1 -- the shape is
% half a Gaussian in linear frequency.
figure(2)
%figure(s1d)
subplot(2,1,2)
semilogx(fv,Make2DFilt(fv,[flo fhi],'highpass'),'b');
set(gca,'box','off','tickdir','out','ylim',[0 1.01]);
xlabel('Radial frequency ( $\text{m}^{-1}$ )');
ylabel('Filter');

% Filter the DEM (D) and return the filtered DEM (Zhp) and filter matrix(F)
[Zhp,F] = SpecFilt2D(D,dx,dy,[flo fhi],'highpass');

%% View filtered DEM

% Display a shaded relief map of the high-pass-filtered data (Fig. 7)
% hp = figure('Name','Fig. 7: High-pass-filtered DEM','NumberTitle','off');
figure(4);
ShadePlot(x,y,Zhp);
hold on
axis equal
axis ij

```

```

title('High-pass-filtered data');
xlabel('Meters');
ylabel('Meters');
hold off

%% Low-pass filter

% What's left over is the low-pass-filtered topography (Fig. 8)
Zlp = Z - Zhp;
%lp = figure('Name','Fig. 8: Low-pass-filtered DEM','NumberTitle','off');
figure(5);
ShadePlot(x,y,Zlp);
hold on
axis equal
axis ij
title('Shaded Relief Low Pass');
xlabel('Meters');
ylabel('Meters');
hold off

figure(6);
hold on
cc = contour(x,y,Zlp);
hold on
axis equal
axis ij
title('Contour Plot Low Pass');
xlabel('Meters');
ylabel('Meters');
clabel(cc);
hold off

% We could also do the low-pass filtering with the following:
% floLP = flo; fhiLP = fhi;
% Zlp = SpecFilt2D(Z,dx,dy,[floLP fhiLP],'lowpass');

```

%%%

Code to classify topography based on curvature

This code was obtained from Martel (2011a), and slightly modified to remove extraneous code and also add a few things relevant to this thesis. This code is used by 'fletcher_curvature_v3' to classify topography based on curvature results obtained from 'curvecode4'. Code was run on MATLAB 2014a Student Edition.

```

function fletcher_topo_shapes_v6(X,Y,Z,tol)
% topo_shapes_fletcher_v1(X,Y,Z,tol)
% function for superimposing contour plots on image plots
% for evaluating shapes of topographic surfaces
%
% tol = tolerance for defining limits of planar surfaces, ridges, etc.

```

```

% Reducing tol shrinks the areas that can be ridges, valleys, or planes and
% increases the regions that can be saddles
% tol = 0.02 seems to work well
%
% Example
%load thin_even_MP.mat
%load thin_even_XP.mat
%load thin_even_YP.mat
%load new_torx
%load new_tory
%
%flo = 1e-3; fhi = 5e-3;
%[Xtf,Ytf,Zhp,Zlp] = topo_filter_short(thin_even_XP,thin_even_YP,thin_even_MP,flo,fhi);
%topo_shapes_cairngorm_v2(Xtf,Ytf,Zlp,0.0015)
%
%figure(5)
%hold on
%plot(new_torx,new_tory,'o','LineWidth',2,'MarkerEdgeColor','k','MarkerSize',10)
%hold off
%
%
%figure(6)
%hold on
%plot(new_torx,new_tory,'o','LineWidth',2,'MarkerEdgeColor','k','MarkerSize',10)
%hold off

% Get Curvature values
[X_,Y_,Z_,K,H,k1,k2,a1,b1,g1,a2,b2,g2,dzx,dzy] = curvcode4(X,Y,Z);

% Set boundaries for small area
M = find (X_(1,:) > -10 & X_(1,:) < 2250); % Change values to slightly less/more than xmin/xmax
N = find (Y_(:,1) > -10 & Y_(:,1) < 3000); % Change values to slightly less/more than ymin/ymax

% Find domes
% Find elements where k1 < -tol and k2 < -tol, and mark them with 1's
D = k1(N,M) < -tol & k2(N,M) < -tol;
Dsum = sum(sum(D));
%[id,id] = find(D);
% Find ridges
% Find elements where |k1| <= tol and k2 < -tol, and mark them with 1's
R = abs(k1(N,M)) <= tol & k2(N,M) < -tol;
Rsum = sum(sum(R));
%[ir,ir] = find(R);
% Find saddles
% Find elements where k1 > tol and k2 < -tol, and mark them with 1's
S = k1(N,M) > tol & k2(N,M) < -tol;
Ssum = sum(sum(S));
%[is,is] = find(S);
% Find planes
% Find elements where |k1| <= tol and |k2| <= tol, and mark them with 1's
P = abs(k1(N,M)) <= tol & abs(k2(N,M)) <= tol;
Psum = sum(sum(P));

```

```

%[ip,ip] = find(P);
% Find valleys
% Find elements where k1 > tol and |k2| <= tol, and mark them with 1's
V = k1(N,M) > tol & abs(k2(N,M)) <= tol;
Vsum = sum(sum(V));
%[iv,jv] = find(V);
% Find bowls
% Find elements where k1 < -tol and k2 < -tol, and mark them with 1's
B = k1(N,M) < -tol & k2(N,M) < -tol;
Bsum = sum(sum(B));
%[ib,jb] = find(B);

%-----

% Plot Curvature Data
figure(1)
clf
imagesc(X_(1,M),Y_(N,1),6*D + 5*R + 4*S + 3*P + 2*V + 1*B)
shading interp;
hold on

% Select Colors, etc.
alpha(0.8); % Sets transparency, with 1 = opaque, 0 = transparent (was .6)
colormap(jet(6));

hcb=colorbar('YTickLabel',{' ','Bowl','Valley','Plane','Saddle *','Ridge *','Dome *',' '});

cRange= caxis; % get the default color map for the data range
caxis(cRange);
hold off;

%-----

% Plot Sheeting Joint Locations
hold on
%fletcher_joints;
lsj = legend('Sheeting Joints Found','show');
set(lsj,...
    'Position',[0.361312053896774 0.134244473838918 0.0860096807527057
0.0268062904174015],...
    'LineWidth',2,...
    'Color',[0.8627 0.8627 0.8627]);

legend1 = legend('Sheeting Joints Found','show');
set(legend1,...
    'Position',[0.358742303529666 0.129780188124632 0.110238755642574
0.0268062904174015],...
    'LineWidth',2,...
    'Color',[0.8627 0.8627 0.8627]);

%-----

% Plot Contour Lines With Smoothed Z Data
[C hT]= contour(X_(N,M),Y_(N,M),Z_(N,M),'k-','LineWidth', 1.0);

```



```

hLines=findobj(gca, 'type', 'line'); % find all the separate lines on contour plot.
set(hLines, 'LineWidth', 1); % and set their width.

% Then reset the color axis according to the range determined above:
caxis(cRange);

% Set axis parameters
axis ij
axis equal;
axis([min(X(1,M)),max(X(1,M)),min(Y(N,1)),max(Y(N,1))]);
xlabel('Meters');
ylabel('Meters');
title('Curvature Results with Filtered Contours');

clabel(C,'manual');

%-----

% Plot histogram of total amount of topographic features

figure(2);
cats = 1:6;
hist_total = [Dsum, Rsum, Ssum, Psum, Vsum, Bsum];
bar(cats, hist_total);
xlabel('Category');
ylabel('Number of Occurrences');

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Code for calculating curvatures, classifying topography, and plotting figures
This script was put together to load in interpolated data, filter the topography using
'fletcher_topo_filter_short_v2', and calculate the curvatures and plot figures using
'fletcher_topo_shapes_v6'. Code was written and run using MATLAB 2014a Student Edition.

% Load Interpolated Data
% griddata Data
%   x = load('gd_800x.txt');
%   y = load('gd_800y.txt');
%   z = load('gd_800z.txt');

% spline2d Data
x = load('s1_800x.txt');
y = load('s1_800y.txt');
z = load('s1_800z.txt');

% Run Smoothing
% Set high and low filter values
flo = 1e-3; fhi = 3.5e-2; % This does well with the fine grid

[X,Y,Zhp,Zlp] = fletcher_topo_filter_short_v2(x,y,z,flo,fhi);

```

```

% Calculate Curvatures
pause
close all
% Set Value for Tolerance
% tol = .0003; or .00035 % This seems to be an ideal tolerance
tol = .00035;
fletcher_topo_shapes_v6(X,Y,Zlp,tol);

% Cover area of No interest
figure(1)
hold on
xnan = [74.45, 1171, 1178, 74.45];
ynan = [1949, 1943, 3026, 3026];
fill(xnan,ynan, 'w');

% Text box with variable values
figure1 = figure(1);
annotation('textbox',...
    [0.359030837004405 0.26625 0.1311 0.1050],...
    'String',{'Low pass Frequency: ',num2str(flo),'High pass Frequency: ', ...
    num2str(fhi),'Tolerance: 0.00035'});

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

References

- Farmin, R. (1937). Hypogene Exfoliation in Rock Masses. *The Journal Of Geology*, 45(6), 625-635. doi:10.1086/624585
- Fletchergranite.com, (2015). Fletcher Granite. Retrieved 9 August 2015, from <http://www.fletchergranite.com/GraniteQuarries/Default.asp?pagecontent=all>
- Gauss, K.F. (1827, translation published in 2005). General Investigations of Curved Surfaces. Dover, Mineola, New York.
- Haneberg, W. (1999). Effects of Valley Incision on the Subsurface State of Stress--Theory and Application to the Rio Grande Valley Near Albuquerque, New Mexico. *Environmental & Engineering Geoscience*, V(1), 117-131. doi:10.2113/gsegeosci.v.1.117
- Holzhausen, G. R. (1977). Sheet structure in rock and some related problems in rock mechanics, Ph.D. dissertation, Dep. of Appl. Earth Sci., Stanford Univ., Stanford, Calif.
- Holzhausen, G. (1989). Origin of sheet structure, 1. Morphology and boundary conditions. *Engineering Geology*, 27(1-4), 225-278. doi:10.1016/0013-7952(89)90035-5
- Jahns, R. (1943). Sheet Structure in Granites: Its Origin and Use as a Measure of Glacial Erosion in New England. *The Journal Of Geology*, 51(2), 71-98. doi:10.1086/625130
- Martel, S. (2006). Effect of topographic curvature on near-surface stresses and application to sheeting joints. *Geophys. Res. Lett.*, 33(1), n/a-n/a. doi:10.1029/2005gl024710
- Martel, S. (2011a). Mechanics of curved surfaces, with application to surface-parallel cracks. *Geophys. Res. Lett.*, 38(20), n/a-n/a. doi:10.1029/2011gl049354
- Martel, S.J. (2011b). Lecture 27: Folds(I). Structural Geology. University of Hawaii at Manoa, Honolulu. Lecture.
- Matthes, F. E. (1930). Geologic history of the Yosemite Valley, U.S. Geological Survey Professional Paper, 160, 137 pp.
- Merk, J. (2010). Weathering and Soils. Geol.umd.edu. Retrieved 10 August 2015, from <http://www.geol.umd.edu/~jmerck/geol100/lectures/12.html>
- Perron, J., Kirchner, J., & Dietrich, W. (2008). Spectral signatures of characteristic spatial scales and nonfractal structure in landscapes. *J. Geophys. Res.*, 113(F4). doi:10.1029/2007jf000866
- Stock, G., & Collins, B. (2014). Reducing Rockfall Risk in Yosemite National Park. *Eos Trans. AGU*, 95(29), 261-263. doi:10.1002/2014eo290002
- Stock, G., Martel, S., Collins, B., & Harp, E. (2012). Progressive failure of sheeted rock slopes: the 2009-2010 Rhombus Wall rock falls in Yosemite Valley, California, USA. *Earth Surf. Process. Landforms*, 37(5), 546-561. doi:10.1002/esp.3192
- Struik, D. (1961). *Lectures on classical differential geometry*. Reading, Mass.: Addison-Wesley Pub. Co.
- Wessel, P., Bercovici, D. (1998). Gridding with Splines in Tension: A Green Function Approach, *Math. Geol.*, 30, 77-93.