

₁ Digital Elevation Models (DEMs) clustering for terrain
₂ modeling

₃ E. R. Stefanescu¹, A.K. Patra¹, and M. Bursik²

₄ ¹Department of Mechanical and Aerospace Engineering, University at
₅ Buffalo, Buffalo, NY 14260

₆ ²Department of Geology, University at Buffalo, Buffalo, NY 14260

₇ May 6, 2013

₈ **Abstract**

₉ We consider the problem of Digital Elevation Models (DEMs) segmentation in
₁₀ homogeneous regions, aiming for identification of plateaux, ridges, small drainages,
₁₁ straight front slopes, valleys, and crests, in order to create ensembles of DEMs. These
₁₂ are then used in a systematic hazard analysis, resulting in a complete and complex haz-
₁₃ ard maps. In the paper we explore and implement a method for segmentation using
₁₄ clustering, that is required / needed when we want to construct a sparse representation
₁₅ of the DEM. The method – spectral clustering, is extensively and successfully used in
₁₆ image segmentation. It is a complex method that accounts for the spatial correlation
₁₇ of the elevation points and has the advantage that it can be used for almost any ap-
₁₈ plication where relationships between topographic features and other components of

19 landscapes are to be assessed. Here, the method is adapted for the case in which each
20 data point has associated range of geomorphometric measures.

21 **1 Introduction**

22 Information about topography is necessary for landscape evaluation, erosion studies, hydrology
23 and geophysical modeling, natural hazard prevention, etc. The classic way to incorporate
24 relief units into a landscape assessment is to delineate them during field survey or using stereo
25 aerial photographs. This approach is relatively time-consuming and the results depend on
26 the subjective decision of the interpreter. Several methods for the creation of landform elements
27 using elevation-derived attributes are described in the literature. Commonly, these
28 techniques developed regions of homogeneity based on common attributes and then classified
29 those regions (or groups of regions) as elements. The most widely used techniques are: self
30 organizing map [Kohn et al., 1995], watershed segmentation [Najman and Schmitt, 1996],
31 support vector machine [Gunn, 1997], segmentation using heuristic rules and fuzzy logic
32 [Sonka et al., 1999], fuzzy K -means classification [Burrough et al., 2000] and object-based
33 image analysis [Carleer et al., 2005]. Many of these techniques have drawbacks, especially
34 when the method relies heavily on hydrological information and requires data-specific knowledge;
35 also these methods don't incorporate autocorrelation between the same attribute at
36 two locations in their models. Digital Elevation Models (DEMs) are digital representations
37 of terrain, and are represented as an array of squared cells (data points/ pixels) with an
38 elevation associated to each data point. They can have different resolutions (5m, 30m, 90m,
39 120m, etc) and can be obtained from various methods (photometry, radar interferometry,
40 laser altimetry, etc.). Usually the size of a DEM varies from tens to hundreds of kilometers
41 which can lead to thousands to millions of grid points.

42 The aim of this paper is to quantify the variation in the output of a computational
43 flow model for block and ash flows, when the model inputs, including the elevation values
44 represented in the DEM, are uncertain or given as a range of possible values. Integrating
45 these variations in the possible flows as a function of input uncertainties provides well-defined
46 data on the probability of hazard at specific locations, i.e., a hazard map [Dalbey et al., 2008].
47 In particular, the focus here is on assessing the influence of DEM uncertainties and propose
48 an improved method of generating ensembles of DEMs.

49 In this context, we would like to implement a more complex model segmentation of a
50 DEM of Mammoth Mountain to create non-overlapping groupings of homogeneous regions.
51 Mammoth Mountain (Fig. 1) is a volcano located in California and it was chosen due to the
ease of obtaining data sets.

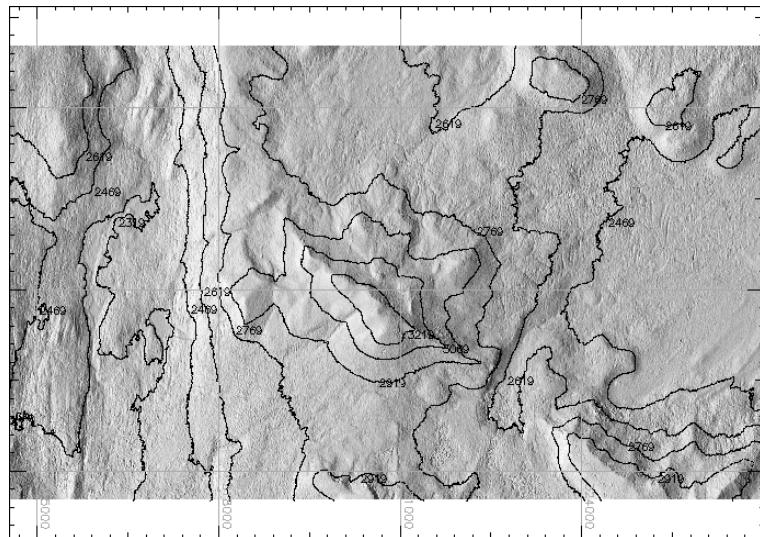


Figure 1: Hillshade plot of the Mammoth Mountain

⁵³ 2 Methodology

⁵⁴ Segmentation is a broad term, covering a wide variety of problems and techniques. Segmen-
⁵⁵ tation methods are based on some data point or region similarity in relation to their local
⁵⁶ neighborhood. A variety of different methods have been proposed for image segmentation
⁵⁷ such as boundary-based segmentation, region-based segmentation and pixel-labeling. One
⁵⁸ view of segmentation is that we are trying to determine which components of the data set
⁵⁹ naturally “belong together”. This is a problem known as *clustering*. Clustering is difficult
⁶⁰ for a number of reasons. Real-life data may contain clusters of varying size and shape,
⁶¹ whose number is unknown in advance. Noise and outliers can further complicate the task
⁶² by connecting separate clusters. Spectral clustering was firstly developed in the context of
⁶³ graph partitioning problems [Donath and Hofmann, 1973], where the problem is to partition
⁶⁴ a weighted graph into disjoint pieces, minimizing the sum of the weights of the edges link-
⁶⁵ ing the disjoint pieces. In the graph the nodes represent the grid points and arcs represent
⁶⁶ affinities (“couplings”) between nearby grid points. The final cluster assignments of the
⁶⁷ dataset can be achieved by optimizing some clustering criteria defined on the graph. The
⁶⁸ criteria of some spectral clustering methods are to optimize some cut value on an undirected
⁶⁹ graph, such as normalized cut [Shi and Malik, 2000], ratio cut [Hagen and Kahng, 1992],
⁷⁰ and min–max cut [Ding et al., 2001].

⁷¹ To be able to perform the segmentation of the DEM in homogeneous regions we need to
⁷² specify a range of geomorphometric measures which can be extracted from the surface. We
⁷³ define a *feature matrix* of DEM attributes, consisting of elevation and first and second deriva-
⁷⁴ tives of elevation (slope, profile curvature and tangential curvature). Slope and curvature
⁷⁵ are easily extracted from a DEM within a Geographical Information System (GIS).

⁷⁶ In the next sections basic methodology for generating ensembles of DEMs using segmen-
⁷⁷ tation is presented, with emphasis on segmentation using spectral clustering. Subsequent

78 sections summarize the TITAN2D flow simulation tool and its use in a systematic hazard
79 analysis. The hazard analysis tool itself uses ensembles of TITAN2D simulations to construct
80 statistical surrogate models of flow outcomes at different locations as a function of model
81 inputs, such as flow volume and pile initial location. Sampling of these surrogates leads to
82 the construction of effective hazard maps that reflect the range of uncertainty in the model
83 inputs.

84 **2.1 Spectral Clustering**

85 A digital representation of a terrain surface is an approximation of reality and is often
86 subject to significant error. The error is usually not known in terms of both magnitude and
87 spatial distribution. There are, in fact large uncertainties associated with the construction
88 of DEMs. DEM vendors generally provide users with a measure of vertical accuracy in the
89 form of the root mean squared error (RMSE) statistic. One key feature of the DEM grid
90 points, which are spatial data, is the autocorrelation of observations in space. Generally,
91 spatial autocorrelation refers to the correlation between the same attribute at two locations.
92 Observations in close spatial proximity tend to be more related than observations at larger
93 distances or separation. Based on this assumption spectral clustering is performed to identify
94 homogenous regions within a DEM.

95 Compared with traditional clustering algorithms, spectral clustering has some advan-
96 tages: can stably detect non-convex patterns and linearly non-separable clusters [Sakai and
97 Imiya, 2009], and can obtain the globally optimal solutions in a continuous domain by
98 eigendecomposition [Archip et al., 2005]. As a discriminative approach, they do not make
99 assumptions about the global structure of data. Instead, local evidence on how likely two
100 data points belong to the same class is first collected and a global decision is then made to
101 divide all data points into disjunct sets according to some criterion. Often, such a criterion

102 can be interpreted in an embedding framework, where the grouping relationships among data
103 points are preserved as much as possible in a lower-dimensional representation. What makes
104 spectral methods appealing is that their global-optima in the relaxed continuous domain
105 are obtained by eigendecomposition. However, to get a discrete solution from eigenvectors
106 often requires solving another clustering problem, albeit in a lower-dimensional space. That
107 is, eigenvectors are treated as geometrical coordinates of a point set. Unfortunately, when
108 the number of grid points (denoted as n) is large, spectral clustering encounters a quadratic
109 resource bottleneck in computing pairwise similarity between n nodes and storing that large
110 matrix. Moreover, the algorithm requires considerable computational time to find the small-
111 est k eigenvalues of a Laplacian matrix. Eigenvalues and eigenvectors are at the heart of
112 spectral clustering algorithms, and in spite of their importance, existing eigensolvers do not
113 scale well. Fast computation schemes for spectral clustering have been proposed by differ-
114 ent authors. They focus on the eigenvector computation of a graph Laplacian defined by
115 a matrix of data similarities. The Krylov subspace methods [Mahadevan, 2008], are iter-
116 ative algorithms for finding leading eigencomponents of a sparse matrix, while Dhillon et al.
117 [2007] assume the availability of the similarity matrix and propose a method that does not
118 use eigenvectors. Fowlkes et al. [2004] propose using the Nyström approximation to avoid
119 calculating the whole similarity matrix. In this paper we are using a method proposed by
120 Song et al. [2008], which parallelize spectral clustering on distributed computers to address
121 resource bottlenecks of both memory use and computation time.

122 2.1.1 Approach

For a given data set $P = \{p_1, \dots, p_n \in R^d\}$, spectral clustering finds a set of data clusters,
 $\{C_1, \dots, C_k \subset P\}$, on the basis of spectral analysis of a similarity graph. Spectral clustering
builds a weighted graph $G(V, E)$, where V represents vertices and E , edges. We represent

each elevation point as a node in the graph G and the links between the adjacent data points will form the edges of the graph. Spectral clustering partitions data points into groups such that the members of a group are similar to each other and dissimilar to data points outside of the group. Given data points, an affinity matrix can be represented by a weighted adjacency matrix $W \in \mathcal{R}^{n \times n}$, where w_{ij} is a measure of the similarity between grid point i and grid point j . The affinity matrix is used to preserve the local structure of the patterns. It expresses the degree of similarity between points, and it must have the following properties: i) non-negative; ii) symmetric; iii) invertible. We have chosen the heat kernel for calculating the affinity matrix, as:

$$W_{ij} = \begin{cases} \exp \frac{-\|F(i)-F(j)\|}{\sigma_F^2} * \exp \frac{-\|x(i)-x(j)\|}{\sigma_x^2}, & \text{if } \|x(i)-x(j)\| \leq r \\ 0, & \text{otherwise,} \end{cases} \quad (1)$$

where $F(i)$ represents the DEM feature vector for node i , and $x(i)$ represents the coordinate location of i^{th} node. σ_F and σ_x are positive tuning parameter that controls the decay of the affinity [Tung et al., 2010]. The graph partitioning can be interpreted as a minimization problem of an objective function. Common objective functions are the ratio cut (Rcut), normalized cut (Ncut) and min–max cut (Mcut) expressed as:

$$Rcut(C_1, \dots, C_k) = \sum_{l=1}^k \frac{(C_l, P \setminus C_l)}{\text{card } C_l} \quad (2)$$

$$Ncut(C_1, \dots, C_k) = \sum_{l=1}^k \frac{(C_l, P \setminus C_l)}{\text{cut}(C_l, P)} \quad (3)$$

and

$$Mcut(C_1, \dots, C_k) = \sum_{l=1}^k \frac{(C_l, P \setminus C_l)}{\text{cut}(C_l, C_l)} \quad (4)$$

Here, $\text{cut}(X, Y)$ is the sum of the edge weights between $\forall p \in X$ and $\forall p \in Y$, $P \setminus C_l$ is the complement of $C_l \subset P$, and $\text{card } C_l$ denotes the number of points in C_l . The degree d_i of

node i is the sum of all edge weights incident on x_i :

$$d_i = \sum_{j=1}^n w_{ij} \quad (5)$$

The *degree matrix* $D \in \mathcal{R}^{n \times n}$ is defined as the diagonal matrix with the degrees d_1, \dots, d_n on the diagonal, while the unnormalized graph Laplacian matrix $L \in \mathcal{R}^{n \times n}$ is defined as [Chung, 1997]:

$$L = D - W \quad (6)$$

Let h_l be an n -dimensional vector indicating the members of the cluster C_l by its binary components. The minimization problem of any objective function in Eqs. 2, 3 and 4 can be rewritten as a trace minimization problem under a constraint on a matrix $H = [h_1 \dots h_k]$:

$$\min_H \text{tr}(H^\top LH) \text{ subject to } H^\top LH = I. \quad (7)$$

The matrix N is equal to I, D or W for Rcut, Ncut or Mcut, respectively [Shi and Malik, 2000, Ng et al., 2002]. The spectral clustering algorithms were derived from the minimization problem in Eq. 7 by relaxing the binary constraint on h_l . The relaxed trace minimization for $H \in \mathcal{R}^{n \times k}$ is the generalized eigenvalue problem [Von Luxburg, 2007]:

$$LH = NH\Lambda \quad (8)$$

The eigenvectors for Ncut and Mcut are identical due to this relaxation, while for Ncut, Eq.8 can be converted into a normal eigenvalue problem:

$$SZ = Z\Delta \quad (9)$$

where

$$S = S_{sym} = D^{-1/2}WD^{-1/2}, \quad Z = D^{1/2}H \text{ and } \Delta = I - \Lambda \quad (10)$$

or

$$S = S_{rw} = D^{-1}W, Z = H \text{ and } \Delta = -\Lambda \quad (11)$$

The above method leads to the normalized graph Laplacians defined as:

$$L_{sym} = I - D^{-1/2}WD^{-1/2} \text{ and } L_{rw} = I - D^{-1}W \quad (12)$$

¹²³ L_{sum} is a symmetric matrix, while L_{rw} it is closely related to a random walk. A more
¹²⁴ detailed description for normalized graph Laplacian can be found in Chung [1997]. The data
¹²⁵ clustering by graph-cut boils down to the eigenvalue decomposition problem of S for finding
¹²⁶ the cluster indicators h_1, \dots, h_k . These eigenvectors induce an embedding of the data points
¹²⁷ in a low-dimensional subspace wherein a partitioning based on the normalized cut (NCut)
¹²⁸ can be used. The solution of the minimization problem can be obtained from the Fiedler
¹²⁹ eigenvector [Yu and Shi, 2003]. The steps involved in DEM segmentation using spectral
clustering are summarized in Figure 2.

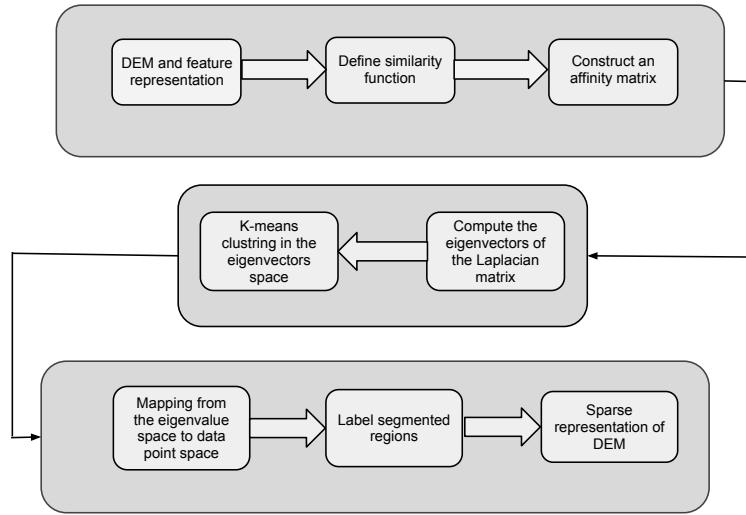


Figure 2: Spectral clustering for DEM segmentation workflow

131 **2.1.2 Parallel implementation**

132 When the number of data points (n) is large, the computational complexity of spectral
133 decompositions can reach $O(n^3)$ (W dense). If the affinity matrix W is define as in Equation
134 1 then its construction takes $\mathcal{O}(n^2d)$ flops, which can be also computationally intensive if the
135 data cardinality n or dimensionality d is large. Chen et al. [2011] investigate approaches for
136 large-scale spectral clustering and propose a parallel implementation, which is also adopted
137 in this paper. The strategy to address the computational and memory difficulties involves
138 distributing n data instances onto p distributed machine node. On each node, parallel
139 spectral clustering (PSC) computes the similarities between local data and the whole set in
140 a way that uses minimal disk I/O. Then PSC stores the eigenvector matrix on distributed
141 nodes to reduce per-node memory use. Together with parallel eigensolver and K -means,
142 PSC archives good speedup with large data sets.

143 **3 DEM ensembles**

When propagating uncertainty in DEMs through a geophysical system, stochastic methods are considered to be an effective way to estimate the probability density function of outputs by addressing uncertainties present in initial conditions and in model approximations. In previous work Stefanescu et al. [2012a] presented a basic methodology for generating ensembles of DEMs representative of the true DEM. Here, we are extended the methodology at the cluster level, by making the assumption that each homogenous regions has its own error model which leads to different random fields. The random fields are used in creating multiple equally likely representations of an actual terrain surface, following the approached suggested by Ehlschlaeger and Goodchild [1994]. A normal distribution (mean of 0.0 and variance of 1.0) of maps or realizations is computed to reproduce the spatial autocorrelation

encountered in the original error surface, filtered using a Gaussian convolution filter, with kernel sizes derived from autocorrelation analysis of the original error surfaces. The random field function derives its spatial dependence from the use of a distance based decay filter function. The following equation is used to generate the random field:

$$Z(\mathcal{U}) = \frac{\sum_v w_{u,v} \epsilon_v}{\sqrt{\sum_v w_{u,v}^2}}, \quad u \in \mathcal{U}, v \in \mathcal{V} \quad (13)$$

$$w_{u,v} = \begin{cases} 1 & : d_{u,v} \leq F \\ \left(1 - \frac{d_{u,v}-F}{D-F}\right)^E & F < d_{u,v} \leq D, u \in \mathcal{U}, v \in \mathcal{V} \\ 0 & : d_{u,v} > D \end{cases} \quad (14)$$

where \mathcal{V} is the set of points potentially influencing points in a given area, \mathcal{U} , $w_{u,v}$ is the spatial autocorrelative effect between points $u \in \mathcal{U}$ and $v \in \mathcal{V}$, ϵ_v is a Gaussian random variable with a mean of 0 and variance of 1, $d_{u,v}$ is the distance between u and v , D is the minimum distance of spatial independence, E is the distance decay exponent, and F the distance at which errors are completely correlated.

A set of random fields are created for each homogenous region/ cluster and are calibrated to the spatial variation of the field being simulated using a correlogram function. This is done by fitting the correlogram and choosing the best descriptive parameters of the random field (the minimum distance of spatial independence, the correlated distance decay exponent and the filter parameter) in a weighted least-square estimator. After running hundreds of tests with multiple combinations of D , E and F , the best random field was found by fitting the error map characteristics such that the sum of least squares difference between an error field's correlogram and the target correlogram is minimized.

Each error realization was added to the “true” DEM indicated as $m(\mathcal{U})$, to generate equally probable realizations of the topography for the error structure of a DEM under

consideration:

$$R(\mathcal{U}) = m(\mathcal{U}) + m(m(\mathcal{T})) + (m(s^2(\mathcal{T})) \cdot \epsilon) \cdot Z(\mathcal{U}) \quad (15)$$

where $R(\mathcal{U})$ is a realization of the elevation dataset $m(\mathcal{U})$, \mathcal{T} is a group of sets of spatially uncorrelated sample points in $m(\mathcal{U})$, and ϵ is a Gaussian random variable with mean 0.0 and variance 1.0. $m(m(\mathcal{T}))$ and variance $m(s^2(\mathcal{T}))$ is mean and variance, respectively, of all sets in \mathcal{T} . $Z(\mathcal{U})$ specifies the random field as defined in Equation 13 for each homogenous region.

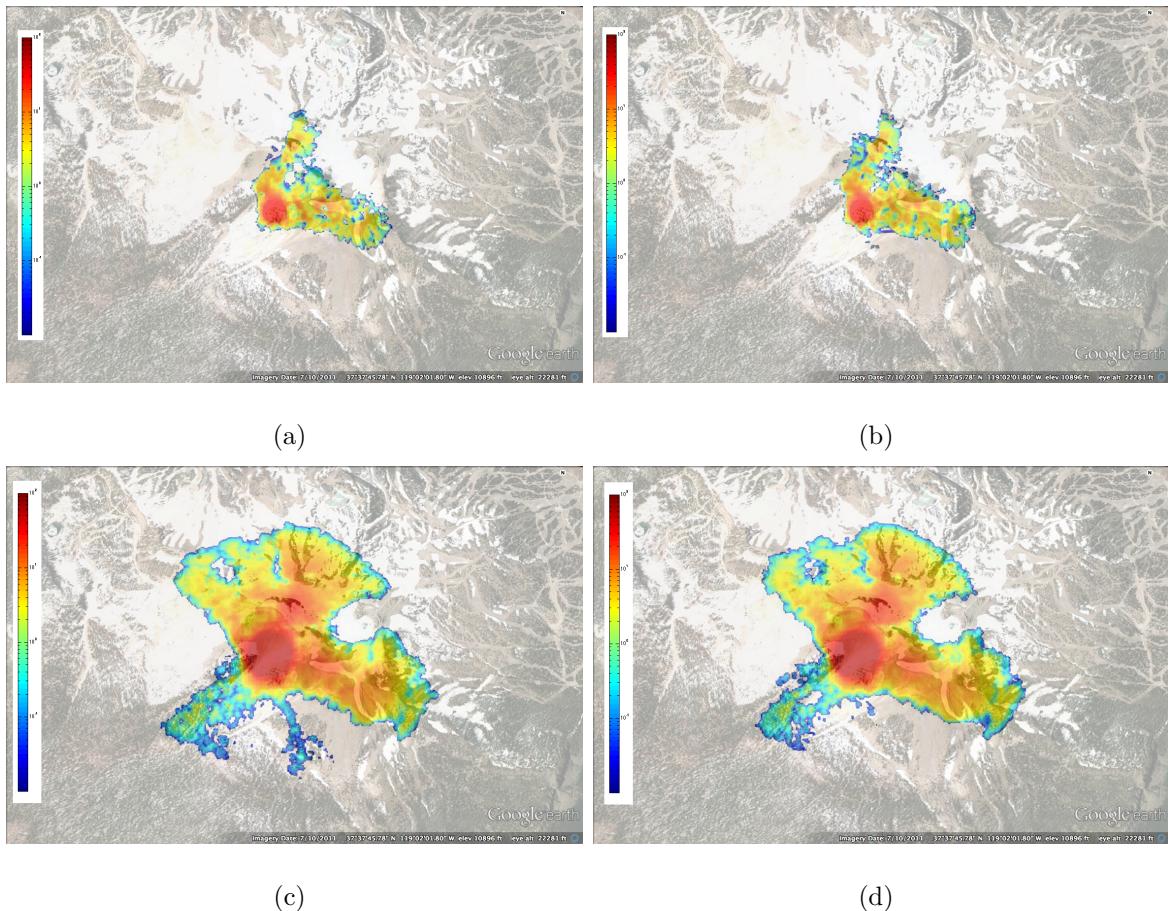


Figure 3: Flow maps for two selected parameters values when DEM was created using a) no cluster b) cluster

₁₆₂ **4 Flow simulator**

₁₆₃ The threat of avalanches, block-and-ash flows, and mud-flows at volcanoes is a major global
₁₆₄ problem. These are complex phenomena and involve physics at multiple spatial and temporal
₁₆₅ scales. Developing satisfactory models and computational simulations of potential debris
₁₆₆ flows on natural terrains and integrating them with appropriate geographical information
₁₆₇ is difficult but extremely necessary. In recent years, many significant advances have been
₁₆₈ made in the modeling and simulation of such flows by taking advantage of new models of
₁₆₉ the physics, new stable and accurate solution schemes for nonlinear hyperbolic systems, and
₁₇₀ the wide availability of high-performance computers.

₁₇₁ Hazard maps are one of the most important instruments [Sheridan et al., 2010] for rep-
₁₇₂ resenting areas of potential inundation by dangerous phenomena at volcanoes. One major
₁₇₃ problem with using model to construct hazard maps is the strong dependence on the outcome
₁₇₄ on the choice of model parameters, such as bed friction, initial volumes and terrain.

₁₇₅ **4.1 TITAN2D**

₁₇₆ TITAN2D [Patra et al., 2005, Sheridan et al., 2005] was developed for modeling dry geo-
₁₇₇ physical granular flows, such as debris avalanches and block and ash flows. The TITAN2D
₁₇₈ code combines numerical simulations of a natural granular flow with digital terrain data.
₁₇₉ It is based on a depth-averaged model for an incompressible granular materia governed by
₁₈₀ Coulomb-type friction interactions [Savage and Hutter, 1989]. The governing equations are
₁₈₁ obtained by applying conservation laws to the incompressible continuum, providing appro-
₁₈₂ priate constitutive modeling assumptions, and then taking advantage of the shallowness of
₁₈₃ the flows (flows are much longer and wider than they are deep) to obtain simpler depth-
₁₈₄ averaged representations [Bursik et al., 2005]. The motion of the material is considered to

185 be gravitationally driven and resisted by both internal and bed friction. The stress boundary
186 conditions are: no stress at the upper free-surface and a Coulomb-like friction law imposed
187 at the interface between the material and the basal surface.

188 A principal feature of TITAN2D is the incorporation of topographical data from geo-
189 graphic information system (GIS) sources into the simulation and grid structure. Topo-
190 graphic data are included in the simulation through a preprocessing routine in which the
191 digital elevation data are imported. TITAN2D performs flow simulations on a DEM of a
192 desired region, the simulation accuracy being highly dependent on the level of the DEM
193 resolution and quality [Stefanescu et al., 2012b].

194 Inputs to the code are the size and location of the initial volume, the internal and bed
195 friction and the DEM. Dalbey et al. [2008] presented several methods for characterizing the
196 effect of input data uncertainty on model output – except DEM, where uncertainty associated
197 with spatial parameters like terrain elevation were not well understood. The output – the
198 flow height at every grid point at every timestep – is a complete description of the mass flow
199 over realistic terrain.

200 We define the stochastic input as $\Omega = (UTM_E, UTM_N, \mathbf{V}, \theta_r)^\top$. UTM_E and UTM_N
201 are the East and North, respectively coordinates (UTM values) of the location of possible
202 vents. These are considered to be uniform distributed around 321095 E and 4166433 N,
203 within 400m radius. \mathbf{V} is the initial volume of material, uniform distributed between 10^5
204 m^3 and $10^{7.5} m^3$. θ_r includes the DEM uncertainty as described in Section 3. We ran 1024
205 flow simulations at design points in the region of the input space. These 1024 design points
206 were chosen according to a Latin hypercube design, which is a space-filling design. This has
207 been proven very successful for all-purpose designs of computer experiment runs since they
208 require relative few design points per input to “fill” the design space [Sacks et al., 1989].

209 **5 Results**

210 The data set available is a 5m DEM, covering an area of $\approx 94km^2$ which results in ≈ 3
211 millions data points. To speed up the computational time, the majority of the analysis was
212 performed on a decimated DEM having a 120m resolution and 6550 data points.

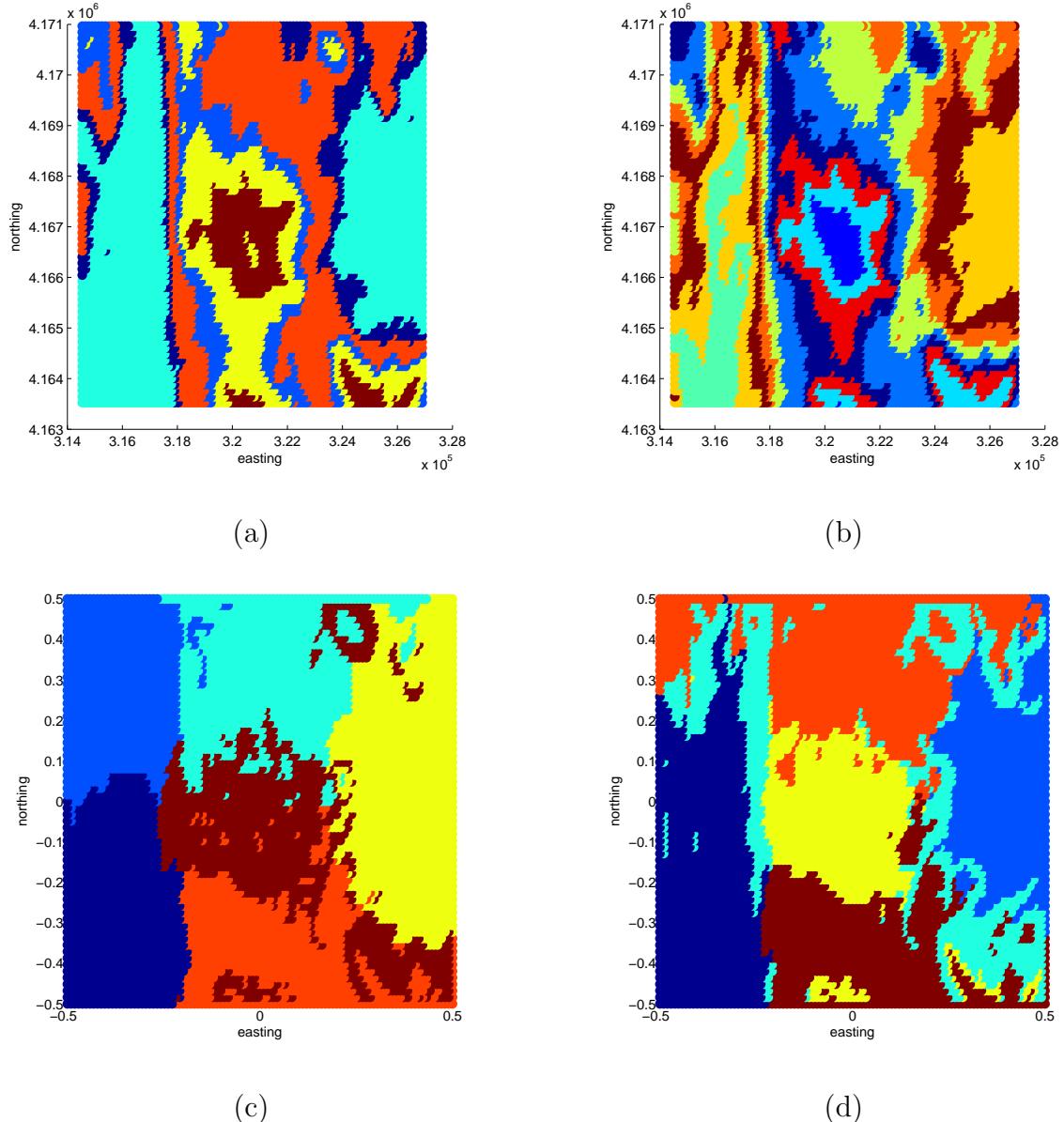


Figure 4: Spectral Clustering for a) $K=6$ – one feature b) $K=20$ – one feature c) $K=6$ where $\sigma_F = 0.8$ and $\sigma_x = 1$ – 4 feature d) $K=6$ $\sigma_F = 0.8$ and $\sigma_x = 0.8$ – 4 feature

213

6 Conclusions

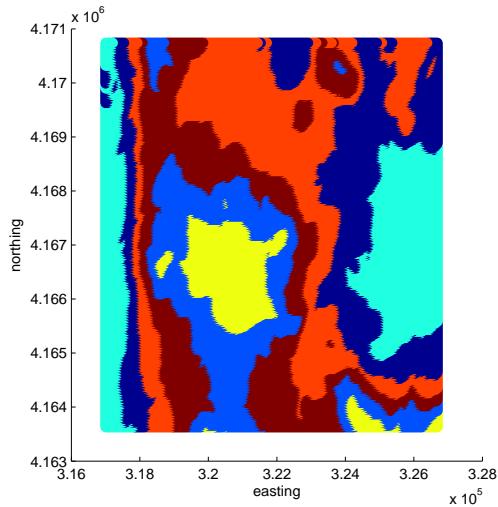


Figure 5: Parallel spectral clustering

214

Acknowledgment

215

References

- 216 N. Archip, P. Rohling, H. Tahmasebpour, and S.K. Warfield. Spectral clustering algorithms
217 for ultrasound image segmentation. *in: MICCAI 2005, Lecture notes in Computer Science*,
218 3750:862–869, 2005.
- 219 P.A. Burrough, P.F.M. van Gaans, and R.A. McMillan. High-resolution landform classifica-
220 tion using fuzzy k-means. *Fuzzy Sets and Systems*, 113:37–52, 2000.
- 221 M. Bursik, A. Patra, E. B. Pitman, C. Nichita, J. L. Macias, R. Saucedo, and O. Girina.
222 Advances in studies of dense volcanic granular flows. *Reports on Progress in Physics*, 68:
223 271–301, 2005. doi:10.1088/0034-4885/68/2/R01.

- 224 A.P. Carleer, O. Debeir, and E. Wolff. Assessment of very high spatial resolution satellite
225 image segmentations. *Photogrammetric Engineering and Remote Sensing*, pages 1285–
226 1294, 2005.
- 227 W.Y. Chen, Y. Song, H. Bai, C.J. Lin, and E.Y. Chang. Parallel spectral clustering in
228 distributed systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*,
229 33 (3):568–586, 2011.
- 230 F.R.K Chung. Spectral graph theory. *CBMS Regional Conference Series in Mathematics*,
231 American Mathematical Society, 92, 1997.
- 232 K. Dalbey. Predictive simulation and model based hazard maps of geophysical mass flows.
233 *PhD thesis, Department of Mechanical and Aerospace Engineering, University at Buffalo*,
234 2009.
- 235 K. Dalbey, A.K. Patra, E.B. Pitman, M.I. Bursik, and M.F. Sheridan. Input uncertainty
236 propagation methods and hazard mapping of geophysical mass flow. *Journal of Geophysical*
237 *Research*, 113:5203–5219, 2008. doi:10.1029/2006JB004471.
- 238 I.S. Dhillon, P. Guan, and B. Kulis. Weighted graph cuts without eigenvectors: A multilevel
239 approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29 (11):
240 1944–1957, 2007.
- 241 C.H.Q Ding, X. He, M. Gu, and H.D. Simon. A minmax cut algorithm for graph partitioning
242 and data clustering. *Proceedings of the IEEE International Conference on Data Mining*
243 (*ICDM*), pages 107–114, 2001.
- 244 W.E. Donath and A.J. Hofmann. Lower bounds for the partitioning of graphs. *IBM Journal*
245 *of Research and Development*, 17:420–425, 1973.

- 246 C. Ehlschlaeger and M.F. Goodchild. Uncertainty in spatial data: Defining, visualizing, and
247 managing data errors. In *Proceedings GIS/LIS'94*, pages 246–253, 1994.
- 248 C. Fowlkes, F. Belongie, F. Chung, and J. Malik. Spectral grouping using the nyström
249 method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26 (2):214–
250 225, 2004.
- 251 M. Goldstein. Bayes linear methods I adjusting beliefs: Concepts and properties. *Part 1 of*
252 *3 of online tutorial, website:*, 2007.
- 253 S.R. Gunn. Support vector machines for classification and regression. *Technical report image*
254 *speech and intelligent group Southampton University*, 1997.
- 255 L. Hagen and A. Kahng. New spectral methods for ratio cut partitioning and clustering.
256 *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 11 (9):
257 1074–1085, 1992.
- 258 J. Kohn, M.S. Suk, and S.M. Bhandarkar. A multilayer self organizing feature map for range
259 image segmentation. *Neural Networks*, 8 (1), 1995.
- 260 S. Mahadevan. Fast spectral learning using Lanczos eigenspace projections. *in AAAI*, pages
261 1472–1475, 2008.
- 262 L. Najman and M. Schmitt. Geodesic saliency of watershed contours and hierarchical seg-
263 mentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(12):
264 1163–1173, 1996.
- 265 A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: analysis and an algorithm. *In*
266 *T. Dietterich, S. Becker, and Z. Ghahramani (Eds.), Advances in Neural Information*
267 *Processing Systems*, 14:849–856, 2002.

- 268 A. Patra, M.I. Bursik, J. Dehn, M. Jones, M. Pavolonis, E. B. Pitman, T. Singh, P. Singla,
269 E.R. Stefanescu, S. Pouget, and P. Webley. Challenges in Developing DDDAS Based
270 Methodology for Volcanic Ash Hazard Analysis Efect of Numerical Weather Prediction
271 Variability and Parameter Estimation. In *submitted to International Conference on Com-*
272 *putational Science*, 2013.
- 273 A.K. Patra, A.C. Bauer, C. Nichita, E.B. Pitman, M. F. Sheridan, M. Bursik, B. Rupp,
274 A. Webber, L. Namikawa, and C. Renschler. Parallel adaptive numerical simulation of dry
275 avalanches over natural terrain. *Journal of Volcanology and Geothermal Research*, 139:
276 1–21, 2005.
- 277 J. Sacks, W.J. Welch, T.J. Mitchell, and H.P. Wynn. Design and analysis of computer
278 experiments. *Statistical Science*, 4(4):409–435, 1989.
- 279 T. Sakai and A. Imiya. Fast spectral clustering with random projection and sampling.
280 *Machine Learning and Data Mining in Pattern Recognition*, pages 372–384, 2009.
- 281 S.B. Savage and K. Hutter. The motion of a finite mass of granular material down a rough
282 incline. *Journal of Fluid Mechanics*, 199:177–215, 1989.
- 283 M.F. Sheridan, A.J. Stinton, A. Patra, E.B. Pitman, A. Bauer, and C.C. Nichita. Evaluating
284 TITAN2D mass-flow model using the 1963 Little Tahoma Peak avalanches, Mount Rainier,
285 Washington. *Journal of Volcanology and Geothermal Research*, 139:275–308, 2005.
- 286 M.F. Sheridan, A.K. Patra, K. Dalbey, and B Hubbard. Probabilistic digital hazard maps for
287 avalanches and massive pyroclastic flows using TITAN2D. *in Groppelli G. and Viereck-*
288 *Goette, L., editors, Stratigraphy and Geology of Volcanic Areas , Geological Society of*
289 *America Special Paper*, 464:281–291, 2010.

- 290 J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern
291 Analysis and Machine Intelligence*, 22 (8):888–905, 2000.
- 292 Y. Song, W.Y. Chen, H. Bai, C.J. Lin, and E.Y. Chang. Parallel spectral clustering. *in:*
293 *Daelemans, W., Goethals, B., Morik, K. (eds.) ECML PKDD 2008, Part II*, 5215:374–389,
294 2008.
- 295 M. Sonka, V. Hlavac, and R. Boyle. Image processing, analysis and machine vision. *PWS*,
296 1999.
- 297 E. R. Stefanescu, M. Bursik, G. Cordoba, K. Dalbey, M. D. Jones, A. K. Patra, D. C. Pieri,
298 E. B. Pitman, and M. F. Sheridan. Digital elevation model uncertainty and hazard analysis
299 using a geophysical flow model. *Proceedings of the Royal Society A: Mathematical, Physical
300 and Engineering Science*, 468(2142):1543–1563, 2012a. doi: 10.1098/rspa.2011.0711. URL
301 <http://rspa.royalsocietypublishing.org/content/468/2142/1543.abstract>.
- 302 E.R. Stefanescu, M. Bursik, and A.K. Patra. Effect of digital elevation model on mohr-
303 coulomb geophysical flow model output. *Natural Hazards*, 62(2):635–656, 2012b. doi:
304 10.1007/s11069-012-0103-y. URL <http://dx.doi.org/10.1007/s11069-012-0103-y>.
- 305 F. Tung, A. Wong, and D.A. Clausi. Enabling scalable spectral clustering for image segmen-
306 tation. *Pattern Recognition*, 43:4069–4076, 2010.
- 307 U. Von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17 (4):395–416,
308 2007.
- 309 S.X. Yu and J. Shi. Multiclass spectral clustering. *In: Proc. Internat. Conf. on Computer
310 Vision*, pages 313–319, 2003.