JOURNAL OF SPATIAL INFORMATION SCIENCE
Number 28 (2024), pp. 3–73
doi:10.5311/JOSIS.2024.28.245

RESEARCH ARTICLE

Locally adjusted curve approximation smoothing of contour polylines

Stephan Miller

Melbourne, Florida, USA

Received: September 25, 2022; returned: December 11, 2022; revised: November 4, 2023; accepted: February 23, 2024.

Abstract: *Numerically interpolated contour lines* (NICL) developed from a high-resolution, accurate digital elevation model (DEM) can be considered the most accurate contour baseline available from surveyed data. Displays of this baseline, however, can exhibit an undesirable angularity and clutter that impedes human understanding of the data and defies the universal appeal of a smooth curve. To achieve smooth, readable lines, the prevailing wisdom today is to either 1) smooth the underlying surface model before creating the lines or 2) use mathematical methods adapted from computer-aided design (CAD) to smooth the lines in x and y. However, these smoothing methods can result in an inaccurate rendition of the original data that undermines their subsequent use in analysis and modeling applications. Upon close inspection, lines generated from such methods can depart from the baseline by more than a definable and acceptable distance in x and y, particularly noticeable in important areas of slope change. Here, we argue that generalized contours have a geometric accuracy requirement that must be observed. We present a novel alternative to the prevailing methods that 1) presents an incremental smoothing method that is a logical complement to Douglas-Poiker thinning; 2) introduces the use of slope as a means of controlling the degree of smoothing locally; 3) produces novel, operational definitions of tests for smoothness and accuracy as well as for line symmetry; and 4) introduces the notion of a topological TIN as a new type of terrain model and a medium for edits of contours. The method only manipulates the baseline contours in x and y. By using only simple proportions of sides and self-similar geometry to smooth the original lines, we at the same time impose computable, dynamic constraints on the smoothing. This combination ensures the accuracy and acceptability of the final contours at scales appropriate for working GIS displays. The result is a *Locally Adjusted Curve Approximation* (LACA) technique for contour line smoothing.

Keywords: terrain modeling, terrain model error, smoothing contour lines, smoothing contour line error, geomorphometry, incremental self-similar line smoothing algorithm, medial axis computation, measures of contour line smoothness and symmetry

1 Introduction

It seems necessary to begin this paper by reaffirming the importance of contour displays to terrain modeling and GIS. Next, we address what increasingly accurate and higher-resolution terrain data means for contour displays. Particularly, we need to discuss the impacts advances in geomorphometry are having on applications of terrain data as an argument supporting greater detail and accuracy in our surface representation and in contour displays.

1.1 The importance of contour displays to terrain modelling and GIS

Numerically Interpolated Contour Lines (NICL) that have been generated by precisely interpolating a series of straight-line segments through a dense and accurate Digital Elevation Model (DEM) provide an undeniably *accurate contour baseline* [10, 35]. Yet such lines are often aesthetically displeasing, erratic in appearance, and hence difficult to read [3, 9, 12]. The process of developing an accurate yet readable contour line representation from high-resolution elevation data is an ongoing area of research in map generalization [3, 7, 9, 27, 33, 36, 37, 52, 60–62]. This research seeks to establish a method of smoothing and enhancement of NICL data that preserves the accuracy and important surface details while ensuring the smoothness and readability of contour displays.

One important impact of the research reported here is that it re-affirms the utility of contour line displays as a vibrant data structure for understanding the surface [4, 18]. The many alternative forms of surface display available in GIS today do not diminish the role of contours as a viable working data structure. This is because the totality of the lines shown in their true relation to one another precisely depict landforms. The shapes of the contours delineate important real-world features of study in the landscape.

To illustrate the importance of these shapes, consider the power of contour displays versus similar displays of either DEMs or Triangulated Irregular Networks (TINs). Viewed from a top-down perspective, neither DEMs nor TINs, the dominant forms of terrain representation in GIS, offer what 2.5D contours can. That is, the display of a matrix of numeric height values representing posts in a DEM is virtually meaningless to a map reader. Likewise, the "spider web" network of a TIN display does little to reveal the structure of the surface. Conversely, contours are readily interpretable by a map reader who can translate the pattern of lines into surface shapes.

To meet their potential, contours must be legible and true to their original shape. The resolution and accuracy of today's LiDAR-derived DEMs underscore this need. As we shall see in section 1.5, the precision and *accuracy* of this data are well on their way to driving changes in Earth science models and our understanding of the world. Contour displays are impacted by this shift in thought and we argue that the accuracy of the smoothed line plays a more important role today than it has in the past.

1.2 Generalization of contours in the context of GIS displays, accurate modern surveys, and emerging trends in geomorphometry

As illustrated by a 1:50,000 scale test conducted for this research (see section 2.3), we believe the method presented here can successfully serve as a front-end to a map production system geared toward topographic map series at large to medium scales. However, it is more

likely to find use in formulating interactive GIS displays of the terrain at larger working scales (i.e., at scales greater than 1:15,000). Even at these scales, the NICL displays require generalization to be most useful as displays for analytic GIS work.

The impact of airborne laser measurements of the terrain is that accuracy and precision have both been elevated to the sublime. Today's terrain models are being captured with single-digit centimeter vertical accuracy [14, 50] and at quarter to half-meter DEM post spacing [26, 47]. At these resolutions, important micro-topographic features are emerging from DEMs. In the past, stream channels from printed topographic maps were used by geomorphologists for analysis. This surface representation gave way to gullies detected by inference from contours on the printed map and verification through costly fieldwork [23]. In turn, this has now given way to multi-temporal detection of rills in the landscape visible in LiDAR imagery [31, 47].

The emergence of micro-features in available terrain data has spurred recent developments in the field of geomorphometry as discussed in the examples given in section 1.5 below. We contend the upshot is that details that had been assumed to be irrelevant noise that could be smoothed away using cartographic license must now be re-evaluated and, as a consequence, more details kept in the smoothed lines.

The approach here is to work the problem of contour legibility in the x and y domain, a marked contrast to widespread methods today which all too quickly resort to smoothing the surface before generating the lines. Surface smoothing approaches, even when coupled with attempts at feature-preserving filter methods, can still degrade important line shapes such as turnbacks and ridgelines to achieve aesthetically pleasing results [29]. The preferred approach presented here offers a proof-of-concept method for calculable thinning and then an algorithm for enhancement of the NICL line by re-constituting sections of it in x and y. The approach balances concern for smoothness with the need for accuracy of line placement and the fidelity of the display to real-world landforms viewable in a GIS context at larger, working scales.

The issue then becomes one of an appropriate degree of generalization of the lines.

1.3 The why, when, and how of contour generalization

Weibel and Dutton [62], in augmenting a list of generalization operators developed by McMaster and Shea [38], describe a framework for generalization in the digital realm that discusses the "why", "when", and "how" of digital cartographic generalization. It is instructive to apply their questions to our work on contour displays.

First, *why* we generalize is easily answered. Despite advances in resolution (or perhaps because of it given the truly discontinuous nature of important parts of the Earth's surface), the surveyed surface is typically described as "rough" and "noisy". Yet the numerically interpolated contours are undoubtedly correct: their method of development accurately reflects the measured surface.

In Figure 1, a generalized version of the NICL lines using thinning and LACA smoothing (shown as black lines at a 1-meter interval) is superimposed over the original display using the method developed (thin red line with vertices). Two basic changes have occurred. First, many small closed contours have been previously eliminated. At the display scale of 1:6,000, these small areas were visually insignificant. Next, the one-meter interval numerically interpolated lines have been thinned and then smoothed using our procedures described fully in Section 3. The smoothed lines are demonstrably easier to interpret, al-

lowing a map reader to concentrate on the *relation among contours* and not be distracted by angular movements occurring *along the individual lines*.

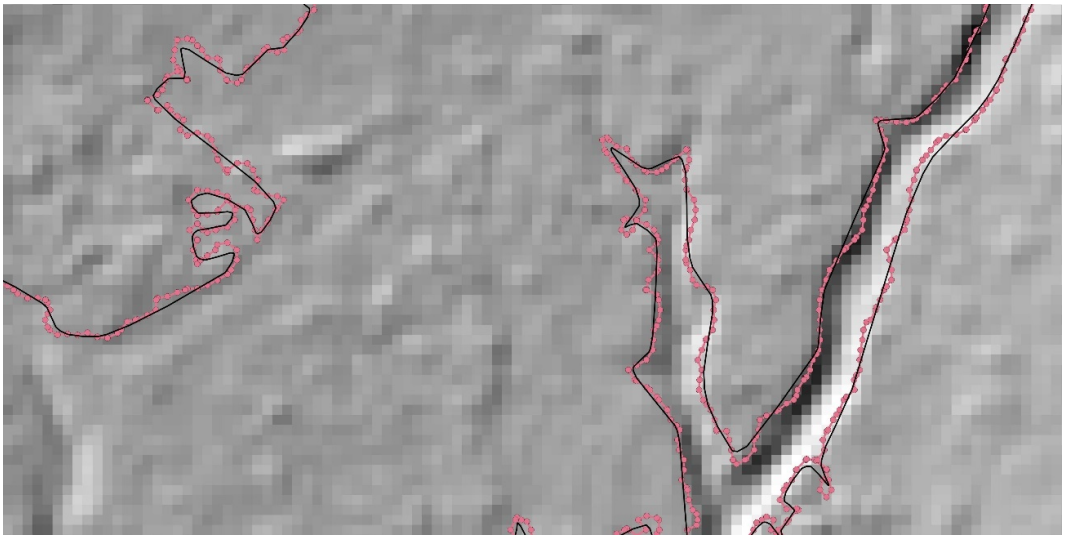


Figure 1: Why We Generalize: Noisy NICL Lines and Smoothed LACA Results for 1-Meter Interval. The irregular red lines and vertices are NICL contours at a one-meter interval (shown to reveal the true nature of the terrain) compared to the black LACA-6K smoothed lines at a one-meter interval.

Second, the *when* of generalization somewhat limits the use of the method described here to larger scales, at which the original vertical error statement of the data still applies [60]. We provide a formula in section 2.4.2.1 for using output scale to drive the line thinning and thus the smoothing that follows. However, when the scale of display is reduced to the point that self-intersections (*coalescence* of adjacent lines or *self-coalescence*) become the norm, we must re-calibrate our thinking. While we do offer the solution given here as a front end to medium-scale topographic map construction (see Sections 2.3 and 4.2), perhaps at this stage we have reached a time to consider the last alternative. That is, to smooth the underlying surface and, to be scrupulous, *necessarily update our vertical accuracy statement* to reflect the lowered confidence in the accuracy of our results. Refer to the test results in Section 4.4 where a revised vertical accuracy statement appropriate for generalized contours is illustrated. Perhaps at this stage, we need to employ a fuller range of generalization techniques as described by Weibel and Dutton [62] and McMaster and Shea [38] and to engage in what Weibel describes as true "cartographic generalization" [60].

Third, as to the *how* of generalization, we employ a limited range of operators: we *eliminate* visually insignificant closed lines; we *simplify* by thinning vertices from the remaining lines by a computable method; and we minimally *displace* and *enhance* the lines by a novel *smoothing* technique that we claim is a logical complement to the Douglas-Peucker thinning method.

The process of "cartographic generalization" is driven by significant scale changes and necessarily employs *intentional* reduction in the accuracy of map data. This leads to the precise point of the dichotomy we face as terrain models today are characterized by extremely

high resolution and accuracy. The accuracy meets the need for greater resolution in support of analysis and modeling efforts as described by two important examples given below in section 1.5. The accurate details of today's data were gained at a considerable price in terms of the mobilization of the new methods of laser data capture and also the attendant problems of data volume that come with increased resolution. Therefore, the details in the data are not to be given up lightly as we try to balance the accuracy and smoothness of contour line displays.

1.4 A problem statement

A clear problem statement for this research is best framed by the USA National Geospatial-Intelligence Agency (NGA) mapping specification [44].

> *Generalization of contours* is necessary since their exact representation would result in *irregular and jagged patterns* which would *hamper readability*. In such instances, the contours are *symmetrically* smoothed, but not to the extent that 1) *the displacement exceeds the geometric accuracy requirement for the map* or 2) *misrepresents the physical characteristics of the terrain* ([44, p.92]—italics and numbered notations are ours).

By *generalization*, we limit ourselves to the aforementioned methods of *elimination*, *thinning*, *displacement*, and *enhancement* of the line. The details present in the data result in the *irregular and jagged patterns* of the numerically interpolated contours that *hamper readability*. The reference to *symmetrical* smoothing is an instruction that the smoothed line should be balanced, perhaps an indirect reference to CAD-based mathematical smoothing techniques in x and y that in the past have appeared overly smoothed and have oscillated wildly to one side or the other of the original line. As we shall see in Sections 3 and 4, we have taken pains to ensure the *symmetry* of the smoothing by working along a medial axis in intervals of the line.

At its core, the proposed method is driven by the realization that generalized contours have a *geometric accuracy requirement*. More exactly, it is argued in Section 2.4 that this accuracy requirement can be tested in x and y and used to both control and assess the degree of generalization.

Finally, in our problem statement, the admonition to not *misrepresent the physical characteristics of the terrain* is an instruction to maintain the shapes of features that are evident in contour displays of the surface. This stricture particularly applies to the drain and ridge features that are vulnerable to methods of surface smoothing.

All too often, turnbacks associated with hydrographic features and ridgelines are subject to oversimplification or elimination when surface smoothing is applied [9, 29, 34]. Alternatively, mathematical functions of smoothing in x and y often produce unnaturally smooth results or require additional control points to condition the line in order to avoid such problems [6, 7].

Both methods are driven by parameters that have a global effect. As Kettunen somewhat gloomily concludes, "parameter settings for universal surface smoothing that fix problems in one area of the surface often produce new errors in other areas and the net result is the same" [29]. The parameters for mathematical smoothing in x and y often require fine-tuning that is difficult to balance from one area to another over the entire area. They often require augmentation points to be added to control the smoothing of the line

and employ numerous iterations to achieve overly-smoothed results in a line [6, see Figures 5 and 6].

1.5 Effects of developments in geomorphometry on contour displays

The availability of high-resolution data and the field of study geomorphometry (the science of quantitative land-surface analysis) are driving a paradigm shift in Earth science [46,49]. To support this shift, contour displays need to be able to retain the meaningful details of the surface [31,49]. The prior emphasis in digital cartography, on smooth surfaces for the production of published, general-purpose topographic maps, should now be extended. It needs to also include an intermediate need for the special purpose, interactive analytic displays of contours in a GIS context using something akin to what Weibel and Burghardt describe as "on-the-fly generalization" [61].

1.5.1 The need to retain important surface breaks and discontinuities

The drive to show greater details in our surface displays is in large part an acknowledgment that the Earth's surface is not continuous as we have long assumed. That it is instead punctuated with slope breaks and discontinuities. As Minár and Evans [41] note in their article on the theoretical axioms of land surface segmentation, the many exceptions to continuity of the surface (i.e., *discontinuities*) need to be retained in the surface model and hence reflected in contour displays. To digress slightly, in fact, we could extend this argument to say that the DEM should be considered sacrosanct and manipulated only for terrain analysis and modeling; that generalization should only be applied to displays in x and y at a target scale as we argue here. Extending this digression, perhaps dangerously further, we could argue that today's high-resolution DEMs contain little or no actual noise, but rather are an accurate rendition of actual irregularities in the surface. At increasingly larger scales for analysis, these irregularities become micro-topographic features that are important to improved analysis, simulation, and modeling [51]. Contours displayed from such a model should honor these discontinuities and not suppress or misrepresent them.

1.5.2 Examples of a paradigm shift: Modelling non-contributing areas in hydrologic models and multi-temporal rills in models of soils erosion

Improved precision, resolution, and accuracy of elevation data are forcing a re-evaluation of many Earth science computational models [21,45,47,50,63]. For example, in hydrologic modeling applications, "pit removal" (i.e., filling in depression areas) is now considered a standard prerequisite technique in virtually all DEM-based watershed delineation models [64]. Perhaps that practice can now be called into question.

DEM filling was designed to alleviate the ambiguity of flow in flat and depressed areas and it accommodates both simple "8D" (8-directional) watershed delineation methods as well as multi-directional flow-routing models [27,28]. Pit removal has always been subject to criticism for its failure to recognize and accommodate "non-contributing areas" [39]. These are areas where the surface legitimately drains to an internal local minimum in the watershed (referred to as "puddles") forming a temporary area of standing water instead of immediately flowing downhill [4,64]. In addition to puddles, any new method must also recognize associated "pour points" or "breach points" in the terrain (i.e., local low points

along the perimeter of depressed areas) where water can ultimately escape the impounded area and continue to flow downhill [39, 64].

The implications of this looming advance for geomorphometry and hydrologic modeling are large. True fully-distributed hydrologic models of overland flow can now be contemplated and dynamically-timed flows from temporary storage in puddles can now be considered in the modeling [47, 64]. This will undoubtedly be an improvement over existing models in terms of the quantity and timing of water released into the connected flowlines in the landscape.

In studies of soil erosion and sediment transport, similar changes in thought are being driven by newly revealed surface details. Details as small as minute turnbacks in contours may be significant in detecting features as small as rills in an application where soil erosion is the point of modeling [31, 64]. Lazzari, in describing the prospects of multi-temporal, high-resolution terrain data, reaches an obvious but challenging conclusion: "Geomorphology [must] develop [a] new geomorphic transport law" [31]. As Dietrich et al. conclude "[w]ith new tools available to obtain high-resolution topographic data and determine rates of processes, the opportunity is now upon us to quantify the processes that are responsible for shaping the earth's surface" [11].

With regard to contour smoothing, all this is not to say that NICL lines require no modifications nor that general-purpose topographic maps have become obsolete. Tunable parameters are still needed to make the numerically interpolated contours legible. Topographic maps will still serve the needs for which they have long been intended and will still be produced, particularly when the need for dramatic scale change becomes evident.

The focus of generalization for working displays in the context of a GIS, however, has been necessarily sharpened with the advent of this new technology. We argue here that minimal line simplification followed by an enhanced form of smoothing is an acceptable level of generalization suitable for analytic terrain work. That is, for what Weibel terms "model generalization". We also suggest that it is better to address the problem of smoothing by displacing contour lines by a calculable amount in x and y rather than 1) uniformly misadjusting the entire DEM to achieve smoother results or 2) relying on mathematical methods that are based on assumptions of the continuity of the cartographic line at the expense of ignoring true breaks and discontinuities in the surface.

1.6 What the method is not

This proof-of-concept contouring method's primary weakness is that it does not consider neighboring lines while thinning and smoothing in its current developed state. The lack of neighborhood information means that it is possible for adjacent thinned or smoothed lines to cross, violating the rules of behavior for contours. This is despite a novel feature of this new method that considers terrain slope in dynamically setting a tolerance for smoothing extent as described later in Section 3.5. Consideration of slope in formulating a smooth contour certainly minimizes the likelihood of crossings but does not remove the risk completely. Later in this case study, we devise a stress test for the possibility of crossing contours by generating contours at a one-half-meter contour interval. The results of the experiment are shown in Section 4.3.

Also, again due to the lack of a neighborhood context, certain basic generalization methods (i.e., *aggregation* or *merging*) that can remove collapsed or coalesced sections of lines due to scale reduction are not currently available with our method. These weaknesses are

acknowledged at appropriate specific points throughout the paper and can be remedied in future work. In Appendices B and C, we speculate at some length on the shape such a solution might take. The real thrust here, however, is to introduce our method of controlled smoothing and to provide comparisons to certain popular alternative methods.

The prospects of adding upgraded checks for significant loss of detail using an advanced TIN data structure and methods suggested by Li and Sui [32] and Weibel [62] and developed by Ware et al. [58] are discussed in the appendices and Section 5 dealing with future research.

1.7 Organization of the paper

The remainder of the paper is organized as follows. Section 2 reviews several principal methods employed today to ensure smooth contours as well as a description of the new method being proposed here. Importantly, in Section 2 and Appendix A, we also outline the proposed methods of testing and the formal criteria to be used in assessing the failure or success of our methods. Section 3 describes the steps in our algorithm and the heart of the geometric operations of line smoothing. Section 4 is an assessment of the success or failure of this method in a series of tests comparing some alternative smoothing methods using the criteria described in Section 2. Finally, Section 5 briefly summarizes conclusions drawn from this work and discusses future directions of work.

2 Methods and materials

We begin this section with a description of the data used for testing as this permits us to deal with specifics as we develop the methods used here. This is followed by a discussion of predominant methods of smoothing for which we offer a typology. Where possible, code has been developed for these alternate methods and implemented in our tests.

2.1 Test data

Our test data consists of two elements, a high-resolution bare earth DEM with one-meter x, y spacing and observations at the ground surface or "bare earth" and, two, contours numerically interpolated from this DEM [1, p.544]. These baseline contours are subject to thinning to elicit the skeletal form of the line at the output scale and this serves as the actual starting point for our method of smoothing and enhancement.

2.1.1 Sources of the data

The posts of the DEM matrix of elevations were developed using an even denser point cloud gathered by a 2017 LiDAR survey of south-central Pennsylvania, USA. This survey has a mean point spacing of 0.7 meters and is representative of LiDAR DEMs in the USA [45].

The elevation data used in this study was developed by the author as a part of a GIS project involving the Gettysburg National Military Park battlefield. The data represents a small portion of the elevation element of the project. It is a one-kilometer by one-kilometer square aligned with Zone 18N Universal Transverse Mercator (UTM) coordinates

expressed in meters. At the prescribed resolution of one meter, the DEM is 1000 rows by 1000 columns for a total of one million points or elevation posts.

2.1.2 Conversion of the scattered LiDAR points to a DEM

The exact methods used to convert the irregular LiDAR observations to a DEM are not clear from the available project documentation [45]. Presumably, this data was dense enough to warrant a simple interpolation method, either nearest-neighbor interpolation or an Inverse Distance Weighted (IDW) interpolation method using neighboring points as described by Watson [59]. The point data were augmented by breaklines, primarily surface drain lines and roads. This ancillary data helped create a better model [21,35,50].

A popular system for processing LiDAR data called LASTools was used for this data. This system includes a LiDAR-to-DEM tool (lasgrid) with many DEM development options [25]. It is not clear exactly which parameters may have been used in this conversion [45].

In general, this data is typical of the high-quality terrain models being generated by LiDAR acquisition methods today [33,45,47,50]. As well, this is likely the type of elevation model that will be used in the foreseeable future and so the process described here seems particularly appropriate for smoothing and generalization of such data.

2.1.3 Vertical error of the DEM

Standard deliverables in terrain modeling projects today include estimates of the vertical error of the elevation survey [54]. The vertical error of the data is important to this study since it can be used to estimate a tolerance for smoothing locally as shown later in Section 3.5.

The "root mean square error" in height ($RMSE_h$), developed from the comparison of DEM elevations and precisely matching surface control points, is used to express the vertical error of the terrain data [53].

For the 2017 DEM data used in this study, the error was determined by comparing 245 field-surveyed control points over the project area to interpolated values from the DEM [45]. For this data, the $RMSE_h$ was found to be an impressive 0.122 meters in open areas of the landscape. For vegetated areas, where the reflective return of the LiDAR is affected by the boles and limbs of the trees, the $RMSE_h$ was degraded to 0.177 meters—still an impressive error figure in comparison to older DEM methods and surveys.

2.2 Prominent approaches to smoothing contour data

We begin this section with a review of some prominent alternative methods of addressing the smoothing issues examined here. Our method is one of many efforts to ensure smooth and accurate contour results. Here we attempt to develop a typology of some prominent methods based on the similarity of underlying characteristics or a common approach. A number of these methods have been selected for testing of accuracy and smoothness in this study to compare to our proposed method.

There are two basic approaches to achieving the desired results. The first category is termed *surface smoothing*. Li and Sui [32] and Gokgoz and Selcuk [18] refer to these types of smoothing as "indirect" versus the second category or "direct" method of *line smoothing*. The indirect category operates on the height values of the underlying DEM before generating the contour lines. By adjusting the height of a DEM center point based on neighboring

elevations of the data, using techniques adapted from image processing, the surface model itself can be smoothed. The contours that result from this process are naturally smoother in x and y by virtue of being interpolated from a smoothed surface model.

The second category, termed *line smoothing*, operates on the x and y coordinates of the line after it has been produced from the unaltered baseline DEM. These data are dense and are usually subjected to thinning in preparation for subsequent smoothing.

Several alternative methods selected from the two categories of smoothing are compared here to our proposed method. We propose a direct method based strictly on the geometry of the lines and driven by two basic parameters, a tolerance for thinning (T_T) and, from this, an automatically derived threshold value for insertion (I_T) of successively finer points in the smoothed line (see Equations 1 and 3 in Sections 2.4.2.1 and 2.4.2.2 respectively).

In the following discussion, surface and line smoothing categories are further subdivided into divisions each with methods that share some basic similarity.

2.2.1 Surface smoothing with multiple iterations (*5X smoothing*)

The most widely applied method of contour smoothing is likely the simple averaging method employed in an *n-by-n* moving window of a DEM. Several variations of this method are widely known. First n may be set to any reasonable odd integer value >= 3 (e.g., 3, 5, 7, 9...), but is normally assigned a default value of 3. Second, the smoothed value may be set equal to a weighted average based on the collection of $n^2 - 1$ neighboring elevation points within the moving window.

Finally, in this surface smoothing method, the number of iterations of smoothing can be varied. In fact, multiple iterations of smoothing are widely recommended in GIS coursework [30], in commercial GIS documentation, and in prominent articles such as Lindsay et al. [34]. It is at this point that many efforts go awry by assigning far too many iterations and suffering a loss of accuracy as a result. For example, the guidance in [30] was 10-15 iterations and, in [34], 5-10 times.

For testing in this study, we developed an intermediary smoothed DEM from the unaltered LiDAR DEM. We set $n = 3$, with weights all defaulted to a value of 1. A "modest" number of five iterations of smoothing was applied. The resulting smoothed intermediate DEM was then used to generate smoothed contours labeled in the tables as *5X Smoothing*.

2.2.2 Numerically interpolated contour lines (NICL)

Contour lines are numerically interpolated from a DEM using a process termed "threading", so called because the process consists of finding the bounding height values in a post neighborhood for the next point in sequence in a developing contour line. This is a conservative linear interpolation method, which is a simple but appropriate technique given the density of the DEM being used. It is worth noting here that the threading technique, combined with the density of the DEM, results in oversampling of the contour line vertices, something to be considered later as we review the results for line smoothness in Section 4.2.

If original blunders or other gross errors that may occur in the LiDAR survey data have been resolved before creating the DEM, we are free to assume that the NICL centerlines derived from the unsmoothed DEM are the most accurate contour baseline available. As such, we will use this data as the baseline for tests involving both the horizontal and vertical measures of error in the smoothed centerlines.

The difficulty of balancing smoothness with accuracy concerns warrants an explicit statement as to the tests to be performed, the criteria to be used to evaluate our method, and the alternative methods investigated here. Details of this testing are relegated to Appendix A. There are eleven data sets developed and used in our analysis which are described in Table A-1. To facilitate the discussion, we present a summary of the data sets using labels that are intended to provide a succinct reference in our discussions. This is followed by Table A-2 describing our intended paired tests using the data sets and Table A-3 which describes four test methods and the criteria for testing.

As a preliminary note to the following discussions, there are three "tolerances", "thresholds", and "factors" employed here that can prove confusing. A "thinning tolerance" (T_T) pertains to controls on point selection and thinning, something undertaken before we begin our smoothing. An "insertion threshold" (I_T) controls the addition of smoothed points into the line during smoothing. A "tension factor" (TF) is a control governing adherence of the smoothed line to the original currently set at a fixed value of 0.4.

2.2.3 Surface smoothing with adaptive DEM smoothing (*Kettunen FPDEMS*)

Recent research has attempted to address the shortcomings of simple surface smoothing methods. These are adaptive smoothing techniques that constitute a second division of surface smoothing methods. Two recent papers are cited as examples here and both present arguments against smoothing of the height data using simple methods [29, 33]. More importantly, however, they also present alternative and more advanced methods of smoothing in height that mitigate some of the significant loss of details such as turnbacks at hydrographic features and switchbacks at ridge protrusions. By identifying significant local inflection points in slope in the DEM, and accounting for this information in their adaptive smoothing approach, these methods are an improvement over multiple iterations of a conventional averaging filter described here.

In a particularly interesting approach, Lindsay et al. [33] coined the term *feature-preserving DEM smoothing* (FPDEMS). As the phrase implies, these are methods that constrain the magnitude of smoothing in slope-sensitive areas of the DEM. Lindsay's method uses distributions of surface normals about a DEM center point to dampen smoothing. Points having a higher variation in normal directions computed relative to their neighbors are smoothed less since they likely correspond to breaks in slope.

An equally ingenious *FPDEMS-type* method by Kettunen et al. [29] also offers assurance that significant features are not completely removed. This method was selected as representative of this division and was implemented and tested for accuracy and smoothness. The method by Kettunen uses the topographic position index (TPI) to control how light and heavy smoothing values are combined locally about a center point in the DEM. The method:

1. computes two intermediate reference DEMs, both conventionally smoothed, one lightly smoothed (1X) and the other heavily smoothed (5X);
2. computes the local difference of elevation of each center point to its neighbors;
3. computes the lowest and highest local difference in elevation values for the original data set; and
4. normalizes the original differences using the overall lowest to highest range.

It then computes a new smoothed DEM value drawing proportionally and inversely proportionally from lightly and heavily smoothed versions of the original DEM respectively. Smoothed contours are generated from the new, composite smoothed DEM.

2.2.4 Surface smoothing using a local patch of elevations (not tested)

A third division of the surface smoothing category is represented by a hybrid method by McCullagh [36] and is presented here to complete a range of available surface smoothing methods. The term "hybrid" is used here to denote the fact that the method combines interpolation and smoothing into a single step.

This method employs local patches of data, either local neighborhoods of DEM data or local triangular patches based on a TIN. Drawing on methods adapted from geological interpolation of subsurface data, the method simultaneously interpolates height at intermediate x, and y positions within the patch and then "laces" (or "threads") the contours through the locally-smoothed patch.

The method is characterized by the use of higher order methods of mathematical interpolation, specifically a "Hermitian bicubic function" is fitted through the four corner points of a local DEM cell (i.e., a local 2 x 2 neighborhood of DEM posts) using a larger neighborhood of overlapping 3 x 3 neighborhoods to compute the equations.

Illustrating this method with a DEM, the first derivative values of the corners are then used to interpolate smooth intermediate values along the neighborhood edges and then broadcast to the predetermined finer subcell locations within an original 2 x 2 neighborhood. The result is a smoothed local surface interior to the initial 2 x 2 neighborhood. These interior points can be threaded directly yielding a portion of a smoothed line that accumulates into a completed line feature. The nature of the method assures that the lines will not cross or otherwise self-intersect. In this regard, this is a notable improvement over our recommended method of LACA smoothing.

McCullagh's approach has an acknowledged drawback that it is computationally intense. More fundamentally, it is based on assumptions regarding the continuity of the lines [40]. Methods making such assumptions tend to produce unnaturally smooth results. This critique applies to this method as well as to a similar line-smoothing technique described below in Section 2.3.1 which follows shortly.

2.3 Line smoothing in x and y

The alternative to the surface smoothing category is directly smoothing the lines themselves in x and y. The first division here involves methods employing higher-order solutions similar to the technique of McCullagh cited above. In general, such techniques are considered appropriate for smooth surfaces in engineering design such as automobile surfaces and in Earth science circumstances where reliable data are sparse as in geologic subsurfaces or for maps of rainfall based on scattered observations at widely separated weather stations.

One of the implications of increased precision and resolution of LiDAR terrain data is that we are working now on a well-defined landscape surface where discontinuities and breaks-in-slope abound. For *surface terrain modeling*, we are no longer working from sparse data conditions where uncertainty in the data justified overly-smoothed contours. The tendency of the CAD-based methods (in fact, any method that advocates too great a smooth-

ness in the lines) is to produce overly smoothed results (see McCullagh [36], Figure 9 and the varying degrees of smoothing in [6]). Interestingly, researchers in human perception have noted that while smooth lines are preferred, there is a fine distinction made between moderate, favored smoothing and excessively smooth lines, which are not preferred [3]. Today, smoothness of this magnitude should be considered a liability for terrain surface modeling.

2.3.1 Line smoothing with a spline technique (tested as *B-Spline Snakes*)

B-Spline Snake functions for line smoothing "deform" (i.e., transform in a positive manner as through pressure) a rough line into a smoothed one. The snakes are so named because, when the processing for deformation solutions are viewed interactively, they resemble writhing snakes.

Guilbert and Saux [22] describe a novel B-Spline Snake function for contour smoothing that seeks to correct the problems with over-smoothing noted here and, to their credit, provides measures of constraint in the process. To this end, they devise a line deformation model that minimizes the sum of the "internal energy" (a measure of the line tension developed from the first derivative of the line) and "external energy" (a measure of the curvature of the line). Again, to their credit, they realize the dangers of topological distortions when smoothing and so they couple the line generalization function with checks for intersections in their process, something our proposed method does not do.

But with its insistence on maintaining the continuity of the line (and also, we believe, erroneous underlying assumptions regarding the degree of smoothness and continuity at the Earth's surface), the B-spline functions, driven by universal parameters, cannot precisely and locally constrain smoothing within as dynamic, tight, calculable bounds as the LACA method does.

In addition, the computations involved in the solution of the B-spline functions and numerous iterations in this method are far more costly and involved than the computations needed for the proposed method [5, 6].

For comparison to our method, and to illustrate the results of higher-order line smoothing, tests were conducted using the *B-Spline Snakes* function available through QGIS 3.16, the open source GIS used in this study.

2.3.2 Line smoothing methods: Using intermediate structures (not tested)

A second division of the line smoothing category worth noting here recognizes the necessity of using advanced intermediate structures to detect and avoid intersections in thinned and smoothed data.

One common thread among these efforts is the use of a TIN to relate the vertices of contour lines [17, 57]. These methods augment the x and y values of the lines by adding surface points at critical junctures where three vertices from a single contour line connect form a "flat triangle", a "problem area", or a "shelf triangle". That is, when an erroneously inferred flat triangles is formed when three points from the same contour (or contours of equal height from nearby lines) are used to create a triangle. In particular, Gokgoz and Gulgen [17] recognize the hidden opportunity in these areas where ridge and drain features can be automatically derived from adjacent flat triangles and then added to the contouring process. Ware [57] proposes a general solution to the problems of "flat triangles" that inter-

polates a smoother surface by introducing data points that force swaps of certain triangle edges, thus replacing flat triangles and improving the model.

Li and Sui [32] describe an integrated approach to the problem of line smoothing that is based on a "natural principle of objective generalization." This approach shares at least two points with the proposed method. In this approach, these researchers imposed a referenced background grid cell based on a minimum "smallest visible object" (i.e., 0.6-0.7 mm square at scale) to develop a generalized version of the contour line. This is somewhat similar to our proposed tolerance for the elimination of small closed polylines based on line thickness and output scale introduced below in Equation 2. The method also eliminates visually insignificant closed lines as proposed in our method. Li and Sui also rely on a local triangulation of adjacent contours to develop contours at a new offset interval when called for (e.g., when changing from a two- to five-meter contour interval).

Gokgoz [16] employs a hybrid approach, combining line thinning and smoothing in his method. By using the accuracy error bands based on planimetric error combined with deviation angles at the vertices, he can select the original line's salient points and stay within the line's original error limits. Gokgoz anticipated several of the features we advance in our proposed method, including using model error to establish a planimetric error band capable of constraining lines. Gokgoz's method is based on incident angles that deviate outside the error band. We use an insertion tolerance and then measure after the fact our success against the error bands.

2.4 Line smoothing using coordinate geometry

The third division within the category of line smoothing includes what we advocate, that is, smoothing using only the coordinate geometry of the line itself. We begin with an introduction relating several past line-smoothing efforts to our proposed method. We follow this with an overview of our proposed method.

2.4.1 Related methods based on coordinate geometry

Other researchers have formulated algorithms that rely principally on the coordinate geometry of the contour lines. In particular, the method described here shares a great deal with one proposed by Dutton where midpoints and a bisecting angle are similar to what is described later in this paper [13]. However, Dutton's method of control of the smoothing process differs, relying on global roundness and smoothness factors which he cleverly related to the fractal dimensions of the line. In our solution, we rely upon a simple, repeatable geometric sequence of interpolation of new points to smooth the line.

Dutton's work is acknowledged as the first to *enhance* smoothed lines [7]. Here we offer enhancement techniques that mimic in reverse the methods of Douglas-Peucker thinning, that is, adding points incrementally and within a point insertion threshold to produce a smoother line. This arrangement is thought to produce an optimal point density for a given level of accuracy and symbolized line thickness of the contours.

The work here, specifically its reliance on simple coordinate geometry and ratios, is perhaps most akin to one of the early algorithms for smoothing developed by George Chaikin and known as the "corner cutting" method [8]. The use of ratios to iteratively create new smoothed positions in the line is common to both approaches. Chaikin applies a constant factor in his method while, as we shall see in the following section for the pro-

posed method, the smoothing ratios used here can be dynamically computed considering local terrain slope by referring to the underlying DEM.

This paper also shares much with a method by Christensen who described an inspiring "ruled surface" method of smoothing the lines [9]. With this method, the proposed one shares the concept of intervals of the line and methods of geometric projection of points in the interval. Instead of solving cubic polynomial equations mathematically, Christensen's "eclectic" geometric method emulates a graphic method developed by engineers at the turn of the last century [9]. Here, wherever possible, we opt for the simplest and most robust methods which are frequently geometrical in nature.

2.4.2 Overview of the proposed smoothing method (*LACA 6K Smoothing*)

In this section of the paper, we provide an overview of our proposed method of line smoothing. We describe the new algorithm and proof-of-concept for smoothing contour lines in x and y that have been previously thinned using an easily computable tolerance based on the output scale. A novel aspect of the new method is that it is capable of adjusting the degree of smoothing on a point-by-point basis according to the local ground slope. A principal strength of this approach is that all the computations are geometrically based, making it simple, fast, and robust.

The smoothing proposed here is composed of three basic steps, all operating in the horizontal domain except for a brief consultation with the DEM to determine local slope variation which can further constrain the line in areas of increasing slope.

First, the Douglas-Peucker method is used to selectively thin the dense, reference *NICL Baseline* contours. Second, our process eliminates visually insignificant, small, closed polyline areas. Finally, our method iteratively adjusts the line and introduces new smoothed points into it until we have ensured a smoothed result that adheres calculably close to the original, accurate line position.

Line thinning of the NICL baseline using Douglas-Peucker. The selective thinning of our process is driven by the output scale and symbolized line thickness which develop the prescribed *thinning tolerance* (T_T) expressed in ground units. A simple formula translates symbolized line thickness to a ground distance needed for thinning as follows.

$$T_T = Scale * Th/1000 \tag{1}$$

where

- $Scale$ is the denominator of the reference output scale
- Th is the line thickness of a symbolized contour line in mm
- 1000 is the constant conversion factor for mm to m

The Douglas-Peucker method of thinning is used here, in part because of its proven durability since its introduction 50 years ago [12] [1, p.429]. As well, the method is based on the geometric properties of the lines alone to select the points with the highest information content, the ones that are most critical to defining the centerline of the data. Those points not selected are thinned leaving a meaningful caricature of the line. The secret to a successful application of this method is to perform only a necessary amount of thinning which, as we demonstrate, is calculable.

Equation 1 sets appropriate tolerances at scale and we will demonstrate the results of an adaptation of this tolerance for 1:50,000 scale later in Section 4.2.4.

Eliminating small contour polygons from the NICL baseline. The original NICL contours consist of 1266 polylines, the vast majority of which are isolated, small closed polylines (or polygons—refer to Figure 14). This is considered to be much of the noise in this LiDAR-derived DEM and the simplification process used here not only reduced vertices from the lines but also, for very small areas, entire lines. The results were telling. Of the 1266 polylines, 1173 were found to be degenerate polygons (i.e., the contour line area had collapsed upon itself in thinning and the computed area is equal to zero) or closed and too small to display at the output scale. Both these types were eliminated, leaving only 93 meaningful candidate lines to be smoothed.

Operationally, the concept of "small" is rooted again in scale as it was when considering the line thickness and the thinning tolerance. At 1:6,000 scale, a minimum sized square area of five times the contour line thickness was used experimentally. A five-by-five-line thickness (5 * 1.2 meters) allows an interior area of three-by-three-line widths, allowing for two-line thicknesses in both x and y for the bounding line itself. In total, the smallest allowable area for a display at 1:6,000 scale is six meters square on the ground or 36 square meters.

$$MIN\_AREA = (T_T * 5)^2 = (1.2 * 5)^2 = 6^2 = 36 \text{ square meters} \tag{2}$$

At a 1:6,000 scale, an "insignificant area" is computed to be 1 mm by 1 mm in size. This is quite small and thought to be a conservative estimate for a minimum size to be used to cull the lines. A closed polyline with an area found to be less than 36 square meters was culled from the list of all polylines for the 1:6,000 scale displays. This simple step eliminates most of the objectionable background clutter from the displays. However, as a side note at this point, one weakness in our approach is that opportunities to aggregate clusters of small, nearby closed contours into one large enough to be portrayed are missed with the currently implemented method [38].

Insertion point threshold. The third step in our smoothing is governed by the *insertion threshold* (I_T) equal to half the T_T value which ensures that the final line is computably smooth enough at the output scale and symbolized line thickness.

$$I_T = T_T * 0.5 \tag{3}$$

In a manner complementary to the Douglas-Peucker line-thinning algorithm, simple geometric relations in the contour line data itself are used to incrementally test and introduce new smoothed points. As we hope to demonstrate in our results, I_T ensures an optimally smooth result using a minimum number of added points.

To illustrate that our method is geometric in nature, we employ frequent use of ratios of sides applied to x and y deltas invoking the principle of similar triangles (reference Figure 10 in Section 3.3). Self-similar replication methods then achieve the desired delicate balance between smoothness and accuracy of the final line data (reference the circles shown in Figure 11 in Section 3.4).

2.5 Tests to be performed and the criteria to be used

Table A-2 in Appendix A enumerates 25 tests and comparisons related to the data sets labeled in Table A-1 representing alternative smoothing methods.

First, we deal with the horizontal accuracy of the lines using each of two methods one of which we have adapted from the literature (Method 1 in Table A-2 in Appendix A). As an additional test, a redundant test to bolster our confidence in results, we have developed an original test method in which we infer horizontal accuracy from departure in the stated height of a smooth point x and y. We compare the observed height accurately interpolated from the original DEM to the stated contour height of the smooth line (see Horizontal Accuracy and Method 2 in Table A-2)

2.5.1 Horizontal accuracy tests (method 1): *NICL Buffers* versus Smoothed Vertices

The tests for horizontal accuracy in our study involve an assessment of the smoothed results for each tested method with the *NICL Buffer 0.6* and *NICL Buffer 1.2*. Instead of using the normal metric of the length of the smooth line lying within the tolerance, here we have calculated the percentage of smoothed points inside the tolerance distance [19]. For a method to pass the test, we must establish that a significant number of our smoothed points lie within 1.2 meters and 0.6 meters of the buffers.

Again, the buffer zones for testing are to be generated using the unthinned *NICL Baseline* center lines minus the small closed shapes that are not considered visibly significant at the output scale of 1:6,000.

Process quality assurance as a testing criteria. Once we determine the percentage of points within epsilon distance to a line, the issue becomes how do we assess the significance of these values? Here, we rely on the literature on process quality assurance and the "Six Sigma" criterion [15]. Quality assurance methods have been successfully applied to a wide range of processes in recent years. For our purposes, we adapt the percentage of "error-free" points (i.e., those falling *inside* the buffers), a simple analytical component of these quality methods, for use in assessing the horizontal accuracy of the smoothed contours [1, p.901].

The distribution of smoothed points about the original line is expected to produce a normally distributed variation and, specifically at the large scale of the test cases, we set a high standard of +/-3 sigma for the points lying within 1.2 meters. Using the percentage proportions for a normal curve, we expect 99.73% (the percentage corresponding to +/- 3 standard deviations or sigma) of the smoothed points to lie within the buffer in ground distance represented by a symbolized contour line thickness set at 0.2 mm. At a scale of 1:6,000, what Guilbert and Saux [22] and Imhof [24] agree is a large to very large scale, we also expect 87% of the points to lie within the more stringent test measure of 0.6 meters. This half value corresponds to +/- 1.5 sigma and this translates to 86.64% of the points. We expect that percentage or greater of the points to lie within half the line thickness of the original numerically interpolated lines.

Computation of the thinning tolerance and insertion threshold. Using Equations 1 and 3 and a line thickness of 0.2 mm, T_T and I_T values are computed as: $T_T = 6,000 * 0.2/1000 = 1.2m$ and $I_T = T_T * 0.5 = 0.6m$.

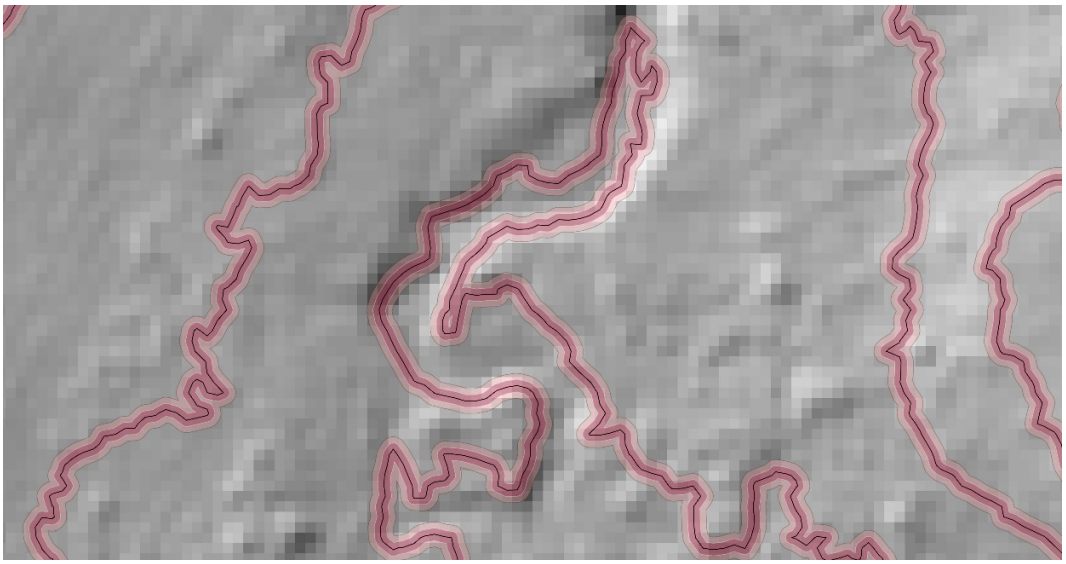


Figure 2: Illustration of the 0.6- and 1.2- meter buffer zones used in horizontal testing for test method 1. The dark red buffer is the *NICL Buffer 0.6m* data set. The outer pink band represents the *NICL Buffer 1.2m* data set. The scale is 1:500.

The former value ensures upfront adequate testing for minimal thinning of the points, while the latter value is the criterion used to assess whether the smoothing should insert an additional smoothed point as it proceeds locally. See the 0.6- and 1.2-meter buffers portrayed in Figure 2.

2.5.2 Horizontal tests (method 1): *NICL Buffers* versus other smoothing methods

In addition to tests of the *LACA 6K Smooth* compared to the *NICL Buffers* (test 1 and 2), Section 4 also includes the results of accuracy for several secondary tests of other smoothing methods discussed here. This includes the following proposed tests:

1. an experiment with B-Spline line smoothing from a GRASS generalization module accessible through the QGIS 3.16 system (*B-Spline Snakes Smoothing*);
2. an experimental implementation of generic surface smoothing of the data by averaging and at 5 iterations of smoothing (*5X Smoothing*); and
3. an experimental implementation of the Kettunen method (*Kettunen FPDEMS*).

All these tests are conducted at a 1:6,000 scale and all will be evaluated against the *NICL Buffers* using the standard 0.6- and 1.2 meter-buffers described earlier (see test descriptions 3 through 8 in Table A-2 in Appendix A).

We have also included a test of our smoothing method using a medium output scale of 1:50,000 (Tests 9 and 10). We expect the simplification necessary for this scale will cause a reduction in the percentage of smoothed points lying within our buffers and the results may include some coalescence of the thinned or smoothed results. However, we hope to successfully demonstrate by our results in Section 4 that we can produce usable contours at a smaller scale with our method. See the results in Section 4.2.4 and 4.2.5.

2.5.3 Horizontal tests (method 2): A test of x and y accuracy by inference using vertical displacement of the smoothed lines

As an added means of assessing the first criterion, the x and y accuracy of the line smoothing, we offer a new method that evaluates the estimated elevations from the DEM at the smoothed x and y positions. See Tests 11-14 in Table A-2 and test type 2 in Table A-3.

The displacement in height of vertices from the stated contour value associated with their parent line is an indirect measure of the success or failure of our method to produce an acceptably accurate display in x and y. Smoothed points lying closer to the *NICL Baseline* are certain to match more closely in height. Importantly, this new method is amenable to hypothesis testing, so it provides a more robust test of our methods than horizontal tests using Method 1, the horizontal accuracy bands alone. Refer to Figure 3.

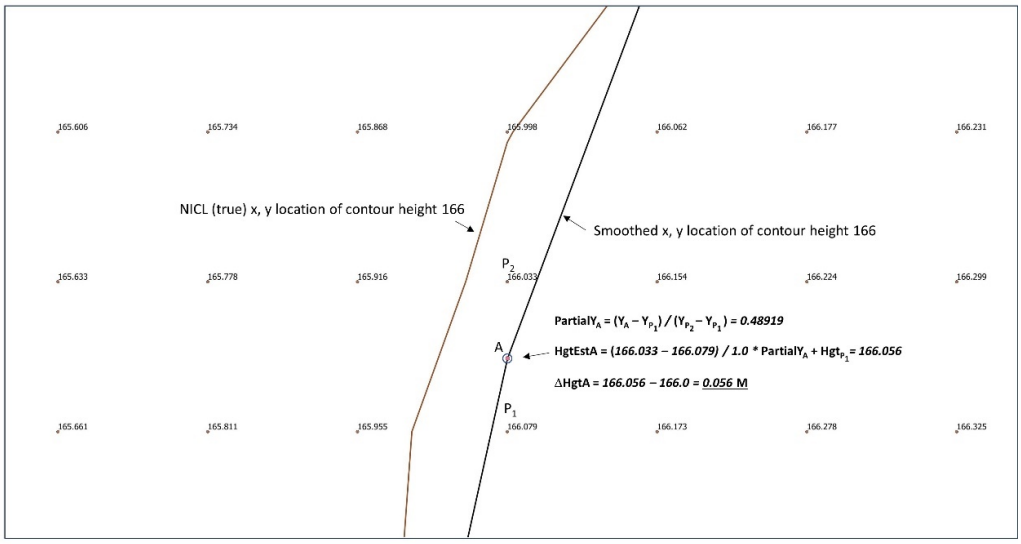


Figure 3: Test method 2: a comparison of stated contour line values in a smoothed line versus height estimates using the original DEM.

Vertices such as point A in the smoothed line in Figure 3 have a *nominal height* of 166 meters. This is the stated contour height of the smoothed line. However, as a test, we can interpolate an estimate of an actual comparative height from the unsmoothed DEM computed in this manner using the x and y of the smoothed vertex. Cumulatively, the deltas in height ($\Delta HgtA$ in the figure) can be summed (mean and standard deviation) and compared with the values of the *NICL Baseline* in a test of means to gauge how closely the lines match. Again, we are inferring a change in x and y from an observed differential in height.

In Figure 3 the brown line represents the true location of the contour at 166 meters (*NICL Baseline*) and the black line is the relocated LACA smoothed line (*LACA 6K Smooth*). According to the smoothed line, point A has a nominal value of exactly 166. But as interpolated precisely from DEM posts, the height is actually 166.056 meters. From this, we

subtract the nominal height of 166.0 rendering a vertical displacement difference of 0.056. This displacement difference ($\Delta HgtA$ in Figure 3) becomes an observation that can be used cumulatively to compute a mean and standard deviation which supports hypothesis testing (see Test Method 2 Hypothesis Testing I in Table A-3 in Appendix A).

2.5.4 Angularity measures of smoothness

The second criterion we will use to define the success of our efforts, that of the contour line's smoothness, is the more difficult of the two criteria to operationally test. For a test, we offer a new measure of angularity that compares successive, overlapping three-point sets in the smoothed line. At each interior vertex of the line (i.e., beginning at the second point in the line and ending at the next to last vertex), we compute the "enclosed angle" (see Figure 4). In this manner, we generate statistics for each vertex for all lines and each of any two compared data sets (i.e., any given smoothing method and the *NICL Baseline*). We then subject the statistics for two compared datasets to hypothesis testing using the mean, standard deviation, and the number of points to gauge the differences between smoothed results and the *NICL Baseline* data. See Test Type 3 in Table A-3 in Appendix A.

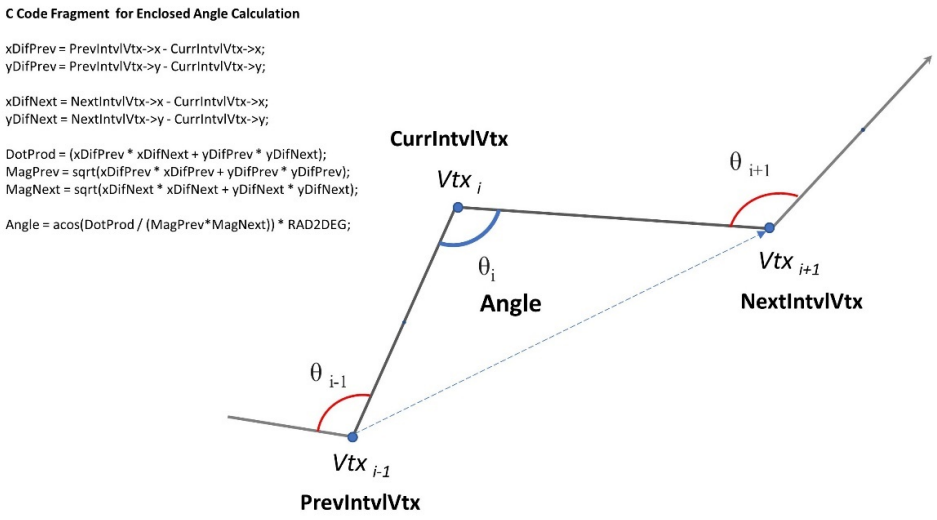


Figure 4: Computation of the enclosed angle at a vertex.

In Figure 4, for a directed line composed of five vertices and three interior angles, and for any current vertex ($CurrIntvlVtx$), the enclosed angle can be computed by forming two vectors hinged at $CurrIntlVtx$ and taking their dot product. As shown in the code fragment depicted in Figure 4, the arccosine of the product of the magnitude of the two vectors (directed from $CurrIntvlVtx$ to $PrevIntvlVtx$ and from $CurrIntvlVtx$ to $NextIntvlVtx$), divided into the dot product renders the unsigned enclosed angle at the $CurrIntvlVtx$ or Vtx_i. The constant conversion factor RAD2DEG translates the angle to degrees from radians which are the normal metric for trigonometric functions.

The color codes for the angles also indicate the orientation (i.e., right or left) of the $CurrIntvlVtx$ to a directed imaginary line connecting $PrevIntvlVtx$ to $NextIntvlVtx$ (see Figure 4). The vertex in the "blue angle" lies to the left of the directed line segment (arbitrarily assigned a negative angle value). The "red angles" lie to the right of the line and are considered to be positive angles.

This convention was originally adopted to provide a simple test of the symmetry of the angles along a line by simply summing the angles and comparing the result to 0.0. This approach produced erratic results and was abandoned, but two new methods for testing symmetry are presented here. See the proposed tests of symmetry in Section 2.5.5.

The frequency distribution of the enclosed angles associated with *NICL Baseline* vertices is presented in Figures 5a and 5b.

Using the convention of positive and negative angles, the frequency distribution of the angles appears to be bi-modal as seen in Figure 5a. For hypothesis testing, the negative side of the distribution is shifted by 360 degrees (expressing the complementary angle) to render a normal-looking curve (see Figure 5b). This transform ensures that all angles are expressed relative to the same (positive) side of the line.

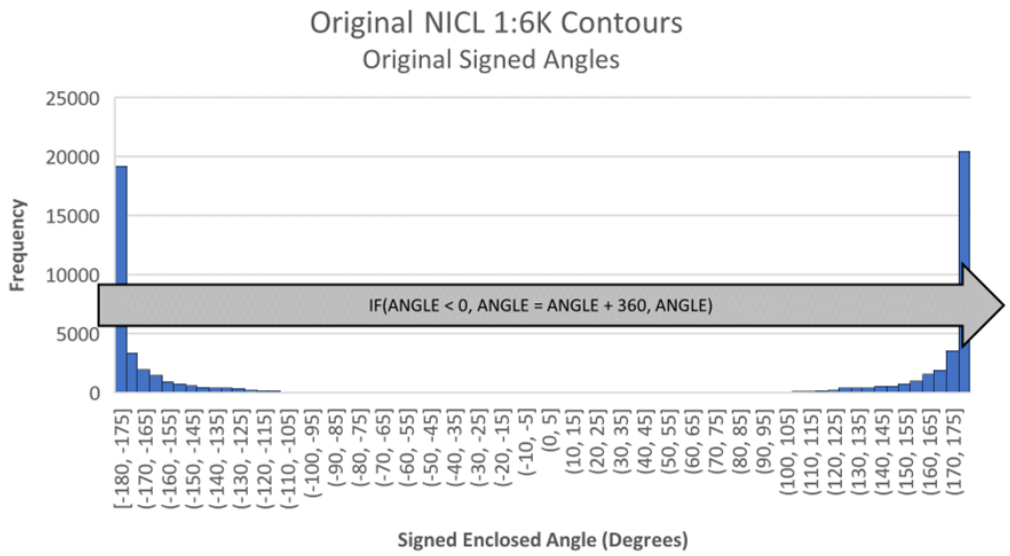
Specifically, in the shifted data, we expect the mean angle to closely approach 180 degrees. The distribution in Figure 5b is a decidedly leptokurtic distribution with a marked peak in frequency for values near the mean. This is thought to be consistent with angles found in a well-behaved and smoothed line and thus, at this point, this is an unusual finding. We expected the *NICL Baseline* data to exhibit high angularity with a larger standard deviation and to be characterized by a higher frequency of acute angles. We did not expect to see the unusual percentage of angles lying close to 180 degrees. To illustrate this concern, the two innermost middle bins of the distribution shown in Figure 5 range from 177.5 to 182.5 degrees and claim 31,000 of 69,700 total points. See further discussion with results in Section 4.2.2.

2.5.5 Test methods for assessing smoothed contour line symmetry

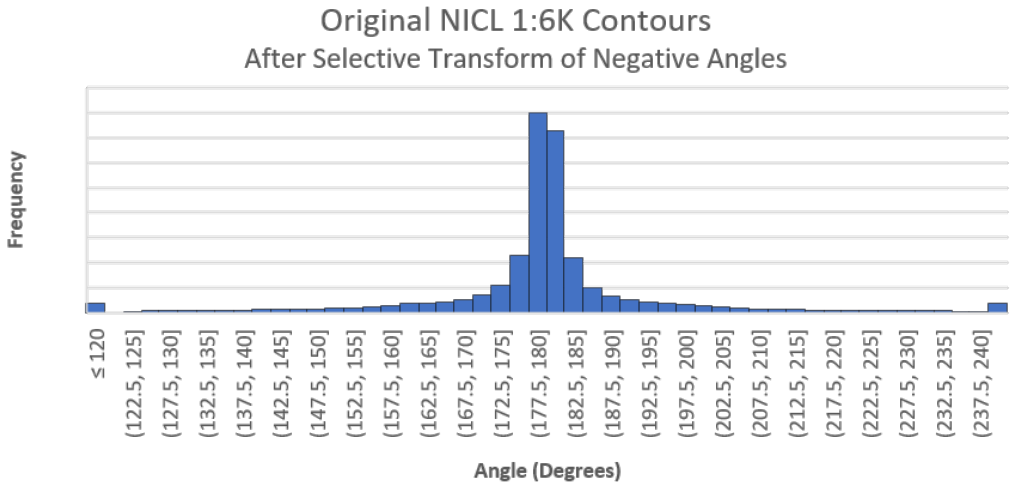
The problem statement in Section 1.4 includes a requirement to *symmetrically* smooth the contour line. Symmetry in a geometric shape is characterized by a composition of exactly corresponding parts arranged facing each other about an axis line. In our case, the shape is a line inspected at each successive overlapping three-point sequence (i.e., an *interval* of the line) and the axis in question is the *medial axis* defined by an *angle bisector* for the interval. Full, operational definitions of these terms follow in Sections 3.1 and 3.2.

Two original methods were devised for assessing symmetry. The first assesses the data from *LACA 6K Smooth* and compares corresponding angles formed between the medial axis point and the vertices introduced by smoothing in the interval (see Figure 6). Since the raw angle values may vary between corresponding vertices, the measure is normalized as a percentage of the larger angle for either side of the interval (see the diagram and computations in Figure 6).

The results of this analysis render a *record of observation* for each matching pair of smoothed points about the medial axis. Taken together, all the observations can be compared using regression analysis between the first and second half of the contour line interval. The coefficient of determination for the relationship can be interpreted as a measure of strength and the slope of the regression line a departure from the ideal symmetry case where Previous % = Next % = 1.0—a case of perfect symmetry.



(a) Distribution of signed angles. The distribution of the signed angles appears to be bimodal. A transformation of the data is in order. The equation is derived from the EXCEL IF statement used to shift negative values to achieve a normal distribution of angles shown below.



(b) The transformed dataset. The negative angles have been shifted 360 degrees to align angles to the positive side of the line and the distribution now appears as a normal though leptokurtic type.

Figure 5: Distribution of enclosed angles for the *NICL Baseline* data.

A second test method for symmetry involves projection of the *LACA 6K Smooth* vertices onto the original *NICL Thin 1.2 m* lines that serve as the starting point for smoothing (see Figure 7). In this method, the interval is again divided into halves: the points lying between

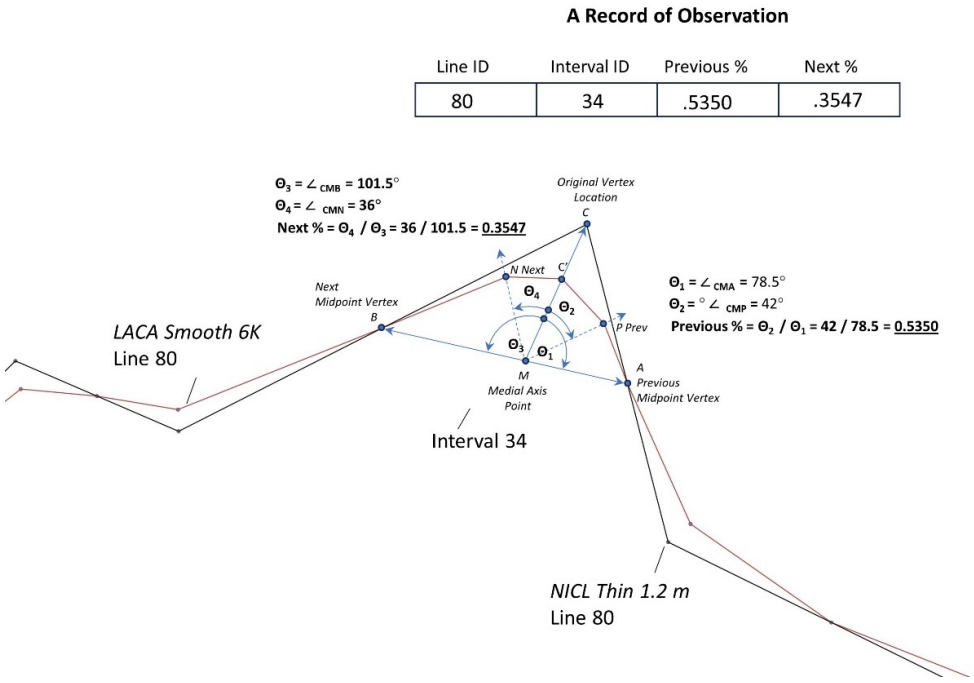


Figure 6: Symmetry assessed by paired comparisons of angles in an interval.

a point A and point C' represent the previous section of the interval while points C' to B represent the next section of the line interval. Corresponding points (i.e., projected P_1 to N_1, projected P_2 to N_2, etc.) provide records of observations, the building blocks for a regression analysis where we can assess the strengthof the symmetry and the departure of the regression line from the ideal.

2.6 Smoothing methods not tested

At the outset, we intended to test at least one method from each of the divisions of smoothing cited here. We fell short of this goal. Despite a surface smoothing Hermitian method being offered in QGIS 3.16, the function did not work using our test data. The method in McCullagh [36] was not sufficiently detailed to allow a direct implementation and, as a result, the hybrid surface smoothing method could not be easily tested.

Also, in the second division of the line smoothing category noted in Section 2.4, the methods of Li and Sui [32] are not tested here. These methods use triangulation of the contour lines and were considered beyond the scope of this effort. Including triangulation with the efforts described here is something to which we aspire. These aspirations are explored by an algorithm proposed in Appendices B and C and in Section 5 in the recommendations for future work.

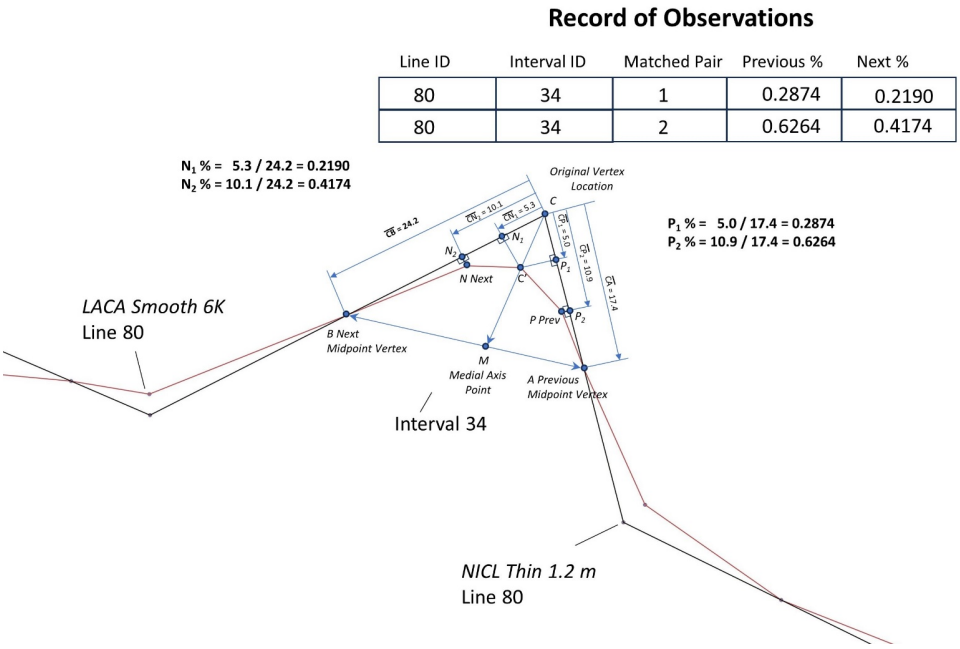


Figure 7: Symmetry assessed by paired comparisons of projected points and lengths in an interval.

Experimentation in the current paper is limited in large part due to the need to develop an adequately flexible TIN structure. The work by Ware et al. [57] describes a *topological TIN* for line generalization, though additional tweaks to the structures may be needed for contours per se (see Appendix A for details). Also, algorithmic modifications would be needed that defy the standard Delaunay criterion for swapping edges (even beyond the standard limits on edge swapping across feature edges in a conventionally *constrained TIN* model). The methods would need to introduce spot height point estimates and derived valley and ridge features at salient points in the terrain much in the manner of Ware [57] and Gokgoz and Gulgen [17].

3 Algorithm for smoothing

This section describes our algorithm developed to smooth and enhance the contours. This includes:

1. the development of an initial interval using midpoints;
2. development of the medial axis of the interval;
3. computing a limiting displacement value for the smoothing;

4. a description of the symmetric smoothing process itself as it devolves into sub-intervals; and
5. the criterion used to insert new, smoother points into the line.

Finally, we illustrate the dynamic nature of the smoothing with an actual example interval from a randomly selected section of a line.

As a note, prototype code was developed for this study in ANSI C using the GNU compiler and debugger, both adapted to run on Windows 10. All the executable code ran in well under 2 seconds on a Dell XPS 8920 with 16 GB of RAM. The times include IO times for debugging prints inside loops and writing the output results to disk.

3.1 Preparing to smooth a line: derivation of the intervals from midpoints

Refer to Figure 8.

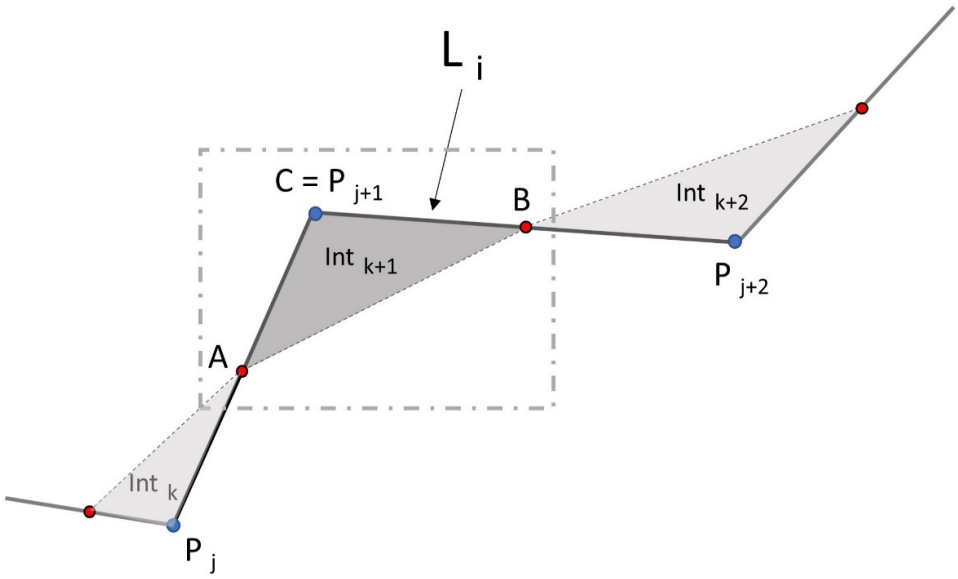


Figure 8: Derivation of the intervals from midpoints.

The algorithm described here is a local method operating on a single interval (Int_{k+1}) of a single line at a time (L_i). An *interval* consists of three consecutive points: two midpoints inserted into the simplified line and shown in red and an enclosed original vertex from L_i in blue (points P_j, P_{j+1}, and P_{j+2} in Figure 8). All intervals of all lines $L_{i,i=1,N}$ are

processed to develop and insert midpoints into a doubly-linked list representation of the polylines.

The midpoints define the maximum extent of an interval and a point of symmetrical transition in the smoothed line between adjacent intervals. The original point in the interval $k+1$, P_{j+1}, is also labeled as vertex C to complete the "triangle" ΔABC which is an alternate representation of the initial interval, Int_{k+1} (see Figure 8).

3.2 Finding the angle bisector for point C and a medial point M on the boundary of the interval along segment AB

Intuitively, the smoothing should be symmetrical within ΔABC and centered on a medial axis of the triangle from point C to a specified medial point M lying on segment AB. The derivation of this medial axis point is described in this section.

Refer to Figure 9. The specific problem may be stated as follows.

Given the interval coordinates of points A and B (segment mid-points introduced into the original line) and C (an original vertex of the line), the segments AB, BC, and CA, and the lengths of these segments, a, b, and c, find the coordinates of a point M, the medial axis point and the intersection of a ray bisecting the angle at C and extending from C through AB.

The individual delta x and y components (Δx and Δy in Figure 9) of segment AB are important and are retained for use in computing the x and y coordinates of point M (equations 10 and 11).

$$\Delta X = X_B - X_A \tag{4}$$

$$\Delta Y = Y_B - Y_A \tag{5}$$

Finally, the coordinates of point M can be developed using the R ratio and the delta x and y for AB as shown in Figure 9. A "straightforward" implementation of this method invokes use of rotations of line segments, and line intersections. As elsewhere in the paper, we seek a simpler and more efficient solution, if possible, preferably one involving ratios applied to x and y differences.

From the Angle Bisector Theorem, we have:

$(AC/AM) = (BC/BM) = K$, always a constant ratio > 1.0. Expressed in terms of lengths, let the distances d and e in Figure 9 represent the distance from A to M and B to M respectively. Restating the Angle Bisector Theorem in terms of these lengths:

$$(c/d) = (b/e) = K \tag{6}$$

This ratio expression of the Angle Bisector Theorem is derived by using algebraic rules for manipulating ratios (see Euclid's definitions for equivalencies at https://mathcs.clarku.edu/~djoyce/elements/bookV/defV14.html). An important expression of the Angle Bisector Theorem can be developed for the sums of the lengths of the sides:

$$(c+b)/(d+e) = K \tag{7}$$

Manipulation of the known lengths of sides c and b can be used to develop a *proportional ratio*, R. This ratio can be applied to the coordinates of segment AB to develop lengths d and e and, more importantly, the coordinates of point M.

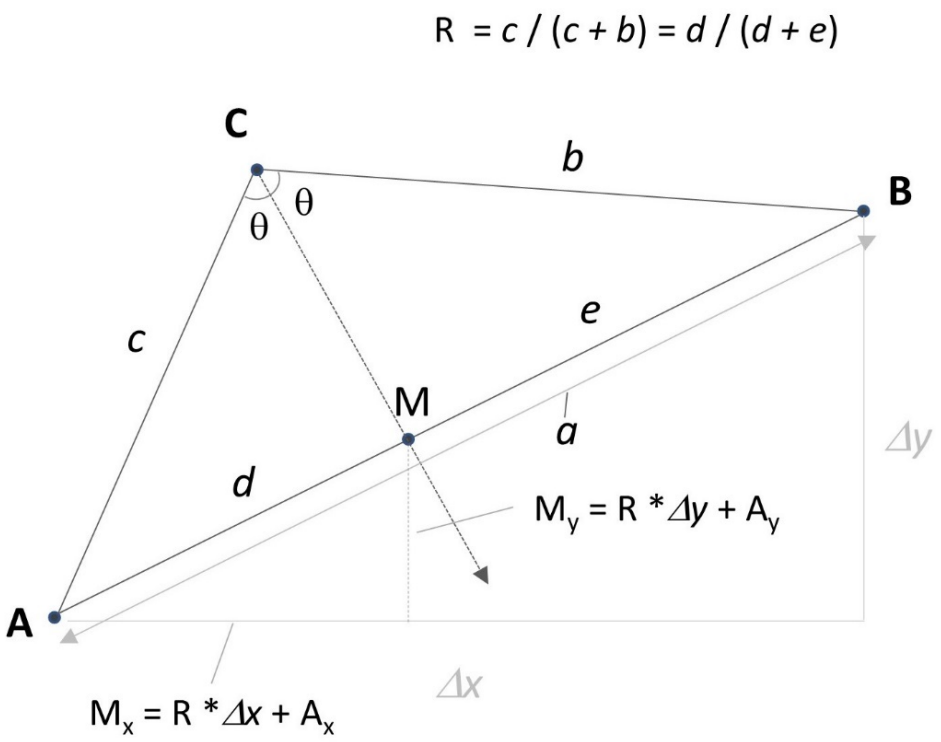


Figure 9: Construction of the medial axis point M and the segment CM. Segment CM divides the angle at Point C into equal portions and is the symmetrical axis for the initial interval ABC.

$$c/(c+b) = d/(d+e) = R \tag{8}$$

The ratio we seek, R, can then be expressed simply as a proportion using *known lengths* c and b alone:

$$R = c/(c+b) \tag{9}$$

The coordinates of M are

$$M_X = (\Delta_X * R) + A_X \tag{10}$$

and

$$M_Y = (\Delta_Y * R) + A_Y \tag{11}$$

The derived point M lies on the medial axis from point C of triangle ABC. This line subdivides the angle ACB, ensuring that the smoothing will be symmetrical within the interval about the medial axis CM. As the original interval develops into finer sub-intervals (e.g., $AC'D$ and $C'EB$), the angles approach 180 degrees while the medial axis remains symmetrical and approaches 90 degrees.

Point C' expresses the limits of smoothing as described below in Figure 10.

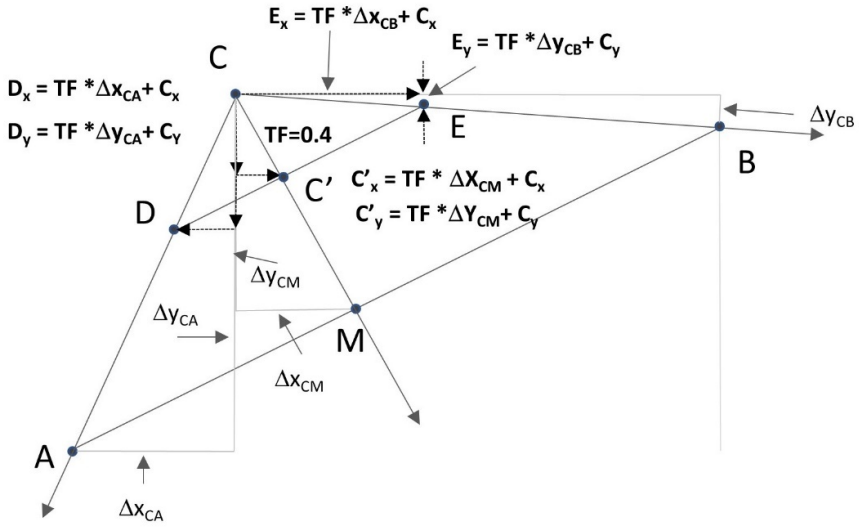


Figure 10: Projections of points C', D, and E in the Line Interval ABC Ratios of triangles and x and y delta values are used to compute offset coordinates. The constant use of the ratio, TF, is the basis for replication of the geometry at the initial interval ABC and in subsequent sub-intervals AC'D and C'BE, etc.

3.3 Using the distance C to point C' as a limit for smoothing

The use of ratios applied to differences in x and y renders many similar right triangles and the ratio R, an expression of the Angle Bisector Theorem, can be used again and again to reveal the coordinates needed for both intermediate and final smoothed points.

If the smoothing were permitted to range as far away from C as point M in an interval, the result would be less than desirable. The line would be decimated and its appearance would approach that of the connected midpoints illustrated in Figure 4. As well as being aesthetically objectionable, this would undermine the likelihood that accuracy would be met and that the acute angles of the line would not be reduced. Some limits to displacement along the line of C to M are needed.

This can be accomplished by setting a factor that is a portion of the total distance of C to M. This "tension factor" TF (to borrow a term from spline smoothing concepts) is a ratio varying from 0 to 1 and is applied to the delta x (ΔxCM) and delta y (Δy_{CM}) from point C to point M to compute a new smoothed or displaced location for the point C designated

in Figure 10 as C'. Similar computations employing the ratio are used to compute the locations of points D and E. See points F, G, H, and I in Figure 11 to reveal the next level of smoothing and what would be symmetric in new sub-intervals such as $AD'F$, $D'C'G$, $C'E'H$, and $E'BI$.

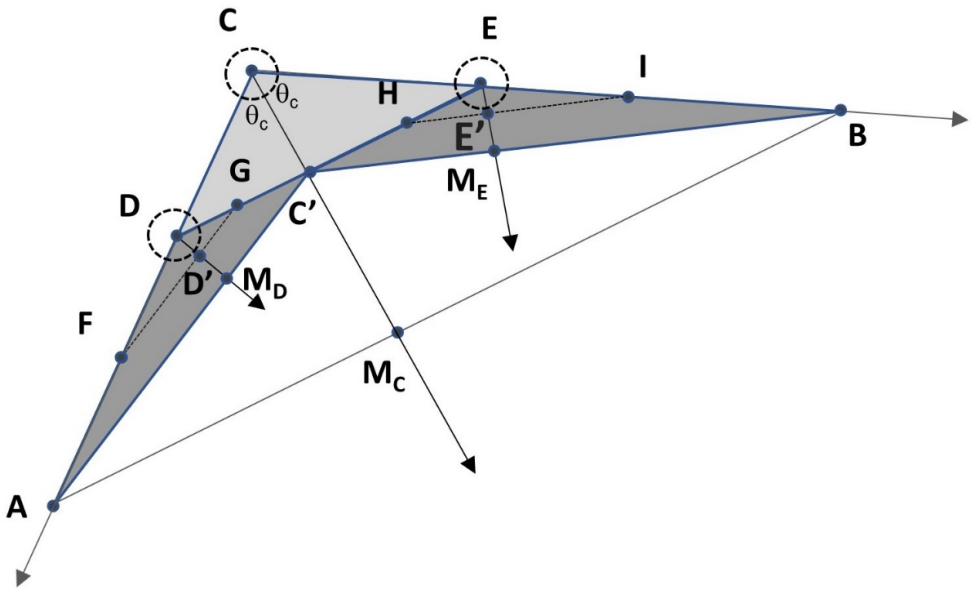


Figure 11: Successive sub-intervals AC'D and C'BE of the original interval ABC. Further sub-interval subdivisions of the line are possible, including AC'D into D'C'G and AD'F and C'BE into C'E'H and E'BI. The process repeats recursively until IT, the insertion threshold, is not exceeded.

This recursive process of geometric replication represents the heart of our smoothing algorithm.

In this strictly geometrical approach to line smoothing, processing an interval of the line involves *displacement* of the point C to position C' along the medial axis towards point M. The amount of smoothing or offset can be controlled in two basic ways. First, a general constraint percentage can be applied to each interval of the line for uniform smoothing. Second, we can develop a modified constraint percentage by considering the local slope of the terrain.

3.4 A description of the symmetrical smoothing process

Refer to Figure 11.

The original interval ABC is recursively divided into two new sub-intervals $AC'D$ and $C'BE$ lying on opposite sides of the medial axis represented by the ray along CM. If this division passes the threshold described in Section 3.5 for the insertion of the new points,

the process repeats with D' and E' and $AD'F$ and $D'C'G$, etc. In this first method of smoothing control, the tension remains constant for all intervals and all lines. A value of TF = 0 means that the line is not smoothed (point C is not moved) and TF = 0.4 means that the line is smoothed to a new location at point C' 40% of the way from point C toward point M. Note that a value of TF = 0.4 worked well under test conditions and is recommended as a default value for this method of smoothing control. Theoretically, the smoothing control could range from 0 to 1.0 with a TF value of 1.0 resulting in the relocation of the line to point M. Aesthetically, 0.4 errs on the side of the medial axis towards C, the original point in the contour line. This is appropriate given our desire to also stay within the bounds of the original line.

Notice that the medial axis angle increases (i.e., approaches 180 degrees) with each sub-interval replication level meaning that the medial axis angle approaches 90 degrees or normal to the original line. This relates back to the issue of *readability* of the contours: smoother lines enhance our ability to visually estimate a normal to the line and to interpolate elevations at points intermediate to consecutive neighboring contours.

The second control method for x, y smoothing is a bit more involved relying on the local vertical slope along the line C to M to dynamically modify the tension factor, TF. It is described in detail in the following section.

3.5 Using terrain model vertical error and local slope to develop a dynamic, locally constrained x, y displacement for smoothing

Refer to Figures 12 and 13.

The objective of this second method of smoothing control is to develop a refined estimate of the displacement tension factor, TF, by dynamically adjusting the percentage change on a local point-by-point basis (i.e., a TF value potentially something less than 0.4 in steep areas and truncated at 0.4 * the length from point C to point MC in flatter areas).

Given an estimate of vertical error of the elevation data, $RMSE_h$, knowledge of the local ground slope can be used to develop a modified percentage estimate for offset for each interval and subsequent sub-intervals discovered during the process of geometric replication.

The tendency for errors in x and y to predict the magnitude of error in height is well-documented [8,20,24]. For a constant slope, a given error in the horizontal produces a correlated vertical offset. That is, as horizontal displacement increases, the vertical displacement in height increases proportionally according to the slope of the land surface [24, p.33] (see Figure 12). As Imhof noted, the uncertainties inherent in repeated measures of the horizontal component of a contour line position reveal what he termed the "zone of mean positional error in contour lines" (MPE) and the arithmetic mean of this distribution is "the most probable location of the contour line" [24, p.33], Imhof's Figure 18.

For our purposes, we need to invert Imhof's problem statement. Typically, in his discussion of the problem, the horizontal error is known and, from this, the vertical error in height is calculated. In our case, we have an estimate of the vertical error in height in the original survey data and need to calculate the allowable horizontal extent of smoothing associated with this error. This can be done based on the surface slope along the medial axis (see Figures 12 and 13).

As an example, a slope of 0.05 (5%), given our data with an $RMSE_h$ = 0.122 meters, produces an allowable x and y offset of 0.122/0.05 = 2.44 meters (see the equation in Figure

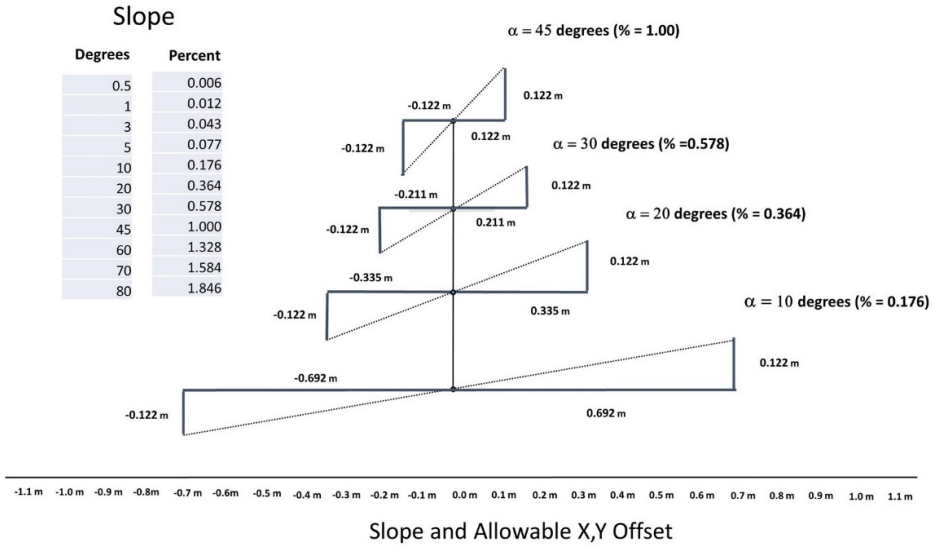


Figure 12: Allowable horizontal offset of displacement for selected slope values. The vertical error for the data test is constant at +/- 0.122 meters. Values of alpha (α) are expressed in degrees followed by their corresponding value expressed in decimal percent ("rise over run"). As the slope decreases, there is, within the limits of the interval midpoints, greater freedom to displace the contour in x and y to achieve smoothness.

13). A slope of 0.01 (1%) produces a more relaxed potential offset of 0.122 / 0.01 = 12.2 meters. Conversely, a percent slope of 1.80 (approximately 80 degrees) produces an offset of only 0.067 meters. In steeper slopes, the smoothed line should lie closer to the original point C. On the other hand, a slope approaching zero implies an infinite allowable x and y offset for the smoothed contour line which we truncate at the maximum allowed tension value of 0.4 (see Equation 11).

A slope approaching 90 degrees implies contours stacked upon one another and complete *coalescence*. Coalescence of contours can indicate problems with the scale and accuracy continuum and may need to be addressed by scale change, contour interval change, or both. Alternatively, perhaps, some allowances should be made for the special display of non-intersecting coalesced contours, perhaps as a *discontinuous feature type* (e.g., a *cliff*, a *retaining wall*, or a *cut bank*). Subtle vertical discontinuities such as this occur in all types of terrain and more frequently than is usually thought. An argument can be made that these features be automatically derived and symbolized in our displays and thereby extend the applicability of a given contour interval that fits most of the terrain.

The allowable offset value is used to adjust the tension value TF by the minimum of either the ratio $xyOffset/Length * 0.4$ or 0.4 itself when the predicted offset exceeds the tension factor for length:

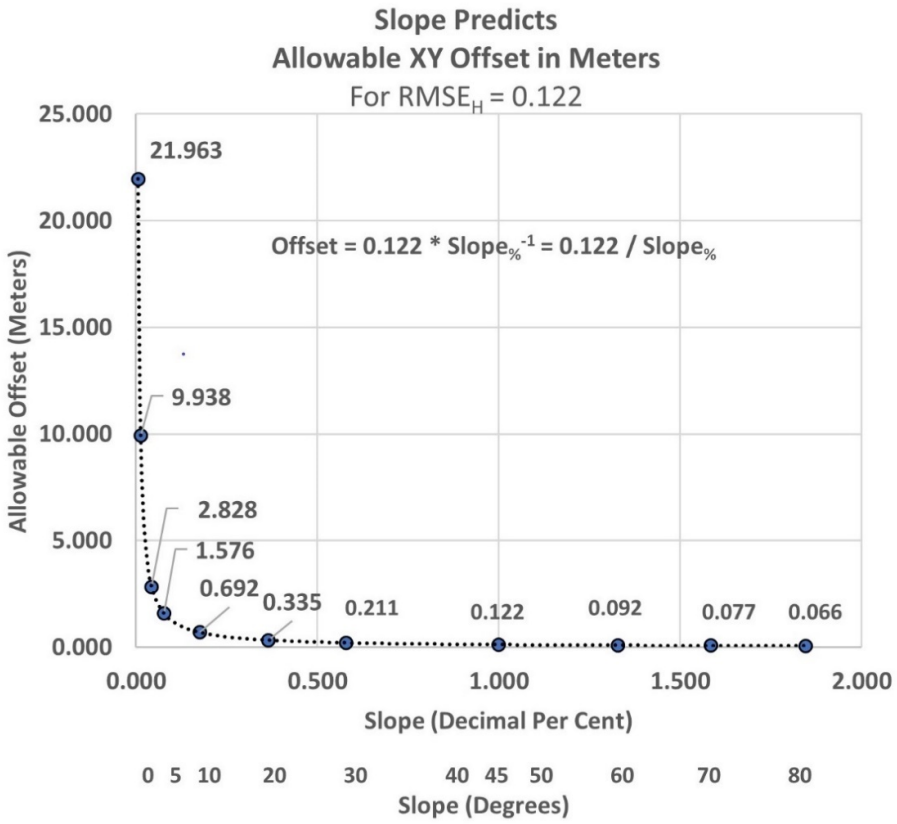


Figure 13: Local slope predicts allowable x, y offset.

$$TF = MIN[(\text{xyOffset}/\text{Length}) * 0.4, 0.4] \tag{12}$$

where

- xyOffset is the potential limit of departure of a smoothed line from the NICL contour in meters along the medial axis computed as
- xyOffset = $RMSE_h * Slope\%^{-1} = RMSE_h / Slope\%$
  - TF = the x, y offset percentage or tension, default value = 0.4
  - $Slope\%$ = the percentage slope as a decimal value (rise over run)
  - $RMSE_h$ = the vertical error associated with the DEM in meters
- Length = $\sqrt{(C_x \breve{} M_x)^2 + (C_y \breve{} M_y)^2}$, the distance of point C to M.

3.6 Insertion threshold based on half the contour line thickness

The process of subdividing the original interval, and displacing the center point in each resulting sub-interval (that is, with points D to D' and E to E', etc. in Figure 11), repeats as long as the respective medial axis distance is greater than half the contour line thickness (0.6 meters for the data and scale used here).

The value of one-half the contour line thickness was derived from the well-known sampling theorem which recommends a minimal interval of half the signal bandwidth to adequately sample a signal [48]. Applying the concept of signal bandwidth to contour lines, the symbolized line width of 0.2 mm at a scale of 1:6,000 is 1.2 meters. This means that we should insert a new prospective smoothed point into the line if the medial distance between points C and C' for any interval (or additional sub-intervals) is less than 0.6 meters. Using this value, we expect to add just enough points in subsequent sub-intervals to ensure the smooth appearance of the output contours at 1:6,000 scale. Here we are adding points that exceed an insertion threshold making the line incrementally smoother—the logical complement to Douglas-Peucker thinning.

3.7 An illustration of smoothing in an interval

To briefly recap, the original *NICL Baseline* data is shown in Figure 14 superimposed over the LiDAR DEM. The *NICL Baseline* is created with small closed contours removed. This original, numerically precise interpolation is the ultimate source of a centerline of the contour as defined by Imhof [24]. (See p. 33 and Imhof's Figure 18). The vertices found in these centerlines are so dense that large scales such as 1:500 to 1:1000 are needed to be able to distinguish the individual vertices in point displays.

The data is then thinned at T_T = 1.2 meters and smoothed using the LACA method to produce the symmetrically smoothed contours visible in Figure 15. A randomly selected area involving a small turnback along Polyline 1233 has been selected to illustrate details in the smoothing process and is shown in Figures 16 and 17. Figure 16 was generated by clipping actual smoothed interval results viewed at a very large working scale in QGIS 3.16. Annotations were added to allow comparison with earlier figures, such as Figures 10 and 11.

4 Results of experiments

Section 4 evaluates the evidence that we have retained the accuracy of the original contour data (Section 4.1) and, at the same time, reduced the angularity and thus improved the smoothness of the lines (Section 4.2). For a general comparison, relative numbers of points for *NICL Baseline*, *NICL Thin 1.2m*, and *LACA 6K Smooth* are given in Table 1. *NICL Thin 1.2m* represents a dramatic reduction in the number of points compared to the *NICL Baseline*, while the reconstituted, smoothed lines in *LACA 6K Smooth* require only one-third the number of the original points.

There are quite a number of tests and comparisons performed on the various data sets used here. Refer to Tables A-1, A-2, and A-3 in Appendix A for the meanings of data set labels, explanatory notes, and criteria for acceptance of the tests conducted.

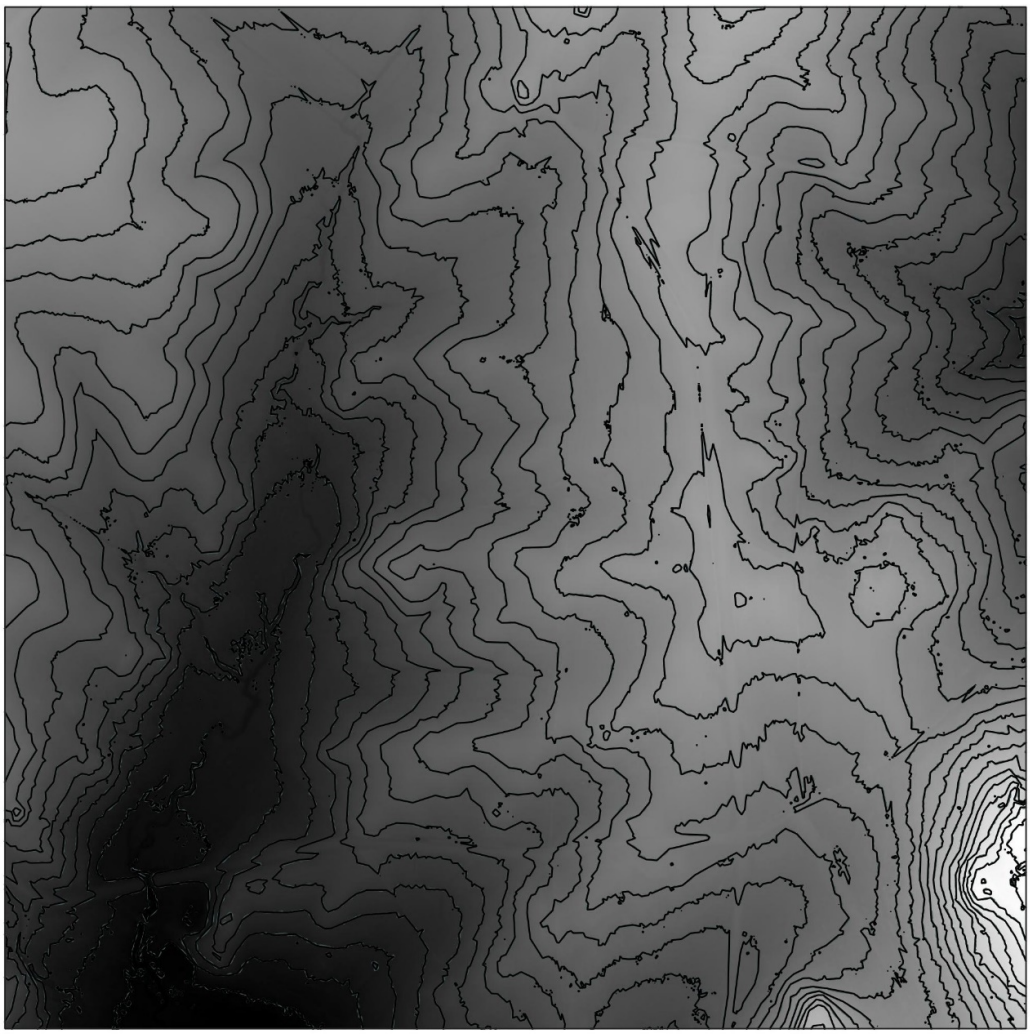


Figure 14: Original DEM with original *NICL Baselines* at 1-meter contour interval. Note the many small closed contours and irregular turnbacks with the blurring of the line as turnbacks in the contour lines are collapsed upon themselves or one another in the display. As shown here, the contour image is displayed at a scale of 1:6200 using the *NICL Baseline* contours before the culling of small areas and before thinning prior to smoothing.

4.1 Horizontal accuracy assessments

The first question to be answered involves the accuracy of the line placement: have we smoothed to an extent that violates the accuracy inherent in the data? To address this, we rely on two test methods simply denoted as Method 1 and Method 2 in Table A-2.

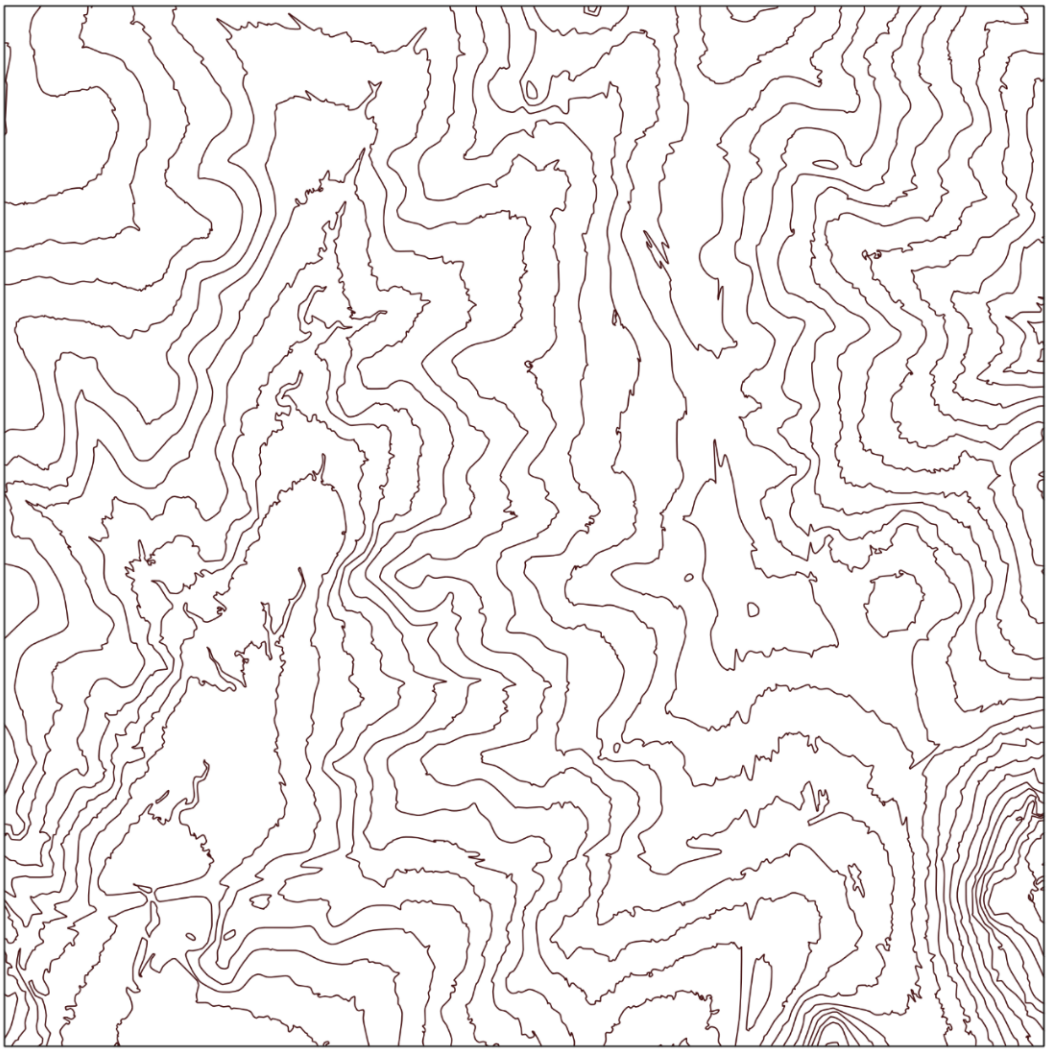


Figure 15: Symmetrically-smoothed contours for *LACA 6K Smooth*. Compare lines to those in Figure 14. The display scale is 1:6200.

First, from past work assessing line placement in GIS, we use the concept of error bands and counts of points lying within an epsilon distance of the *NICL Baseline* contours [19, 42, 43]. This is Method 1 of the Horizontal Accuracy Tests.

Second, to strengthen our argument regarding x and y accuracy, a new test method is proposed in Section 4.1.2. The method is specific to contours and infers horizontal accuracy from the vertical displacement of smoothed contour vertices compared to height estimates from the original DEM (see Section 2.5.3, Figure 3, and Tests 11-15 in Table A-2). This is Method 2 of the Horizontal Accuracy tests.

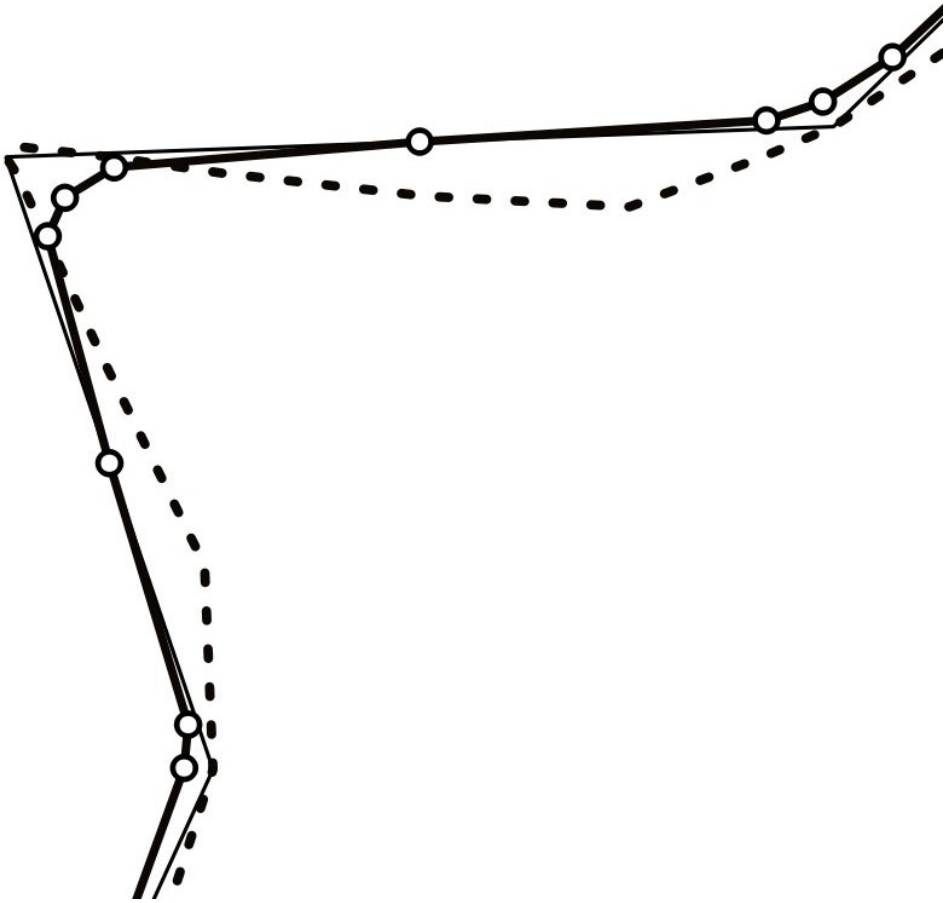


Figure 16: *NICL Baseline*, *NICL Thin 1.2m*, and *LACA 6K Smooth* results. The dashed line represents the *NICL Baseline*; the thin black line represents the *NICL Thin 1.2m* data; and the solid black line with white vertices represents the results of the *LACA 6K Smooth* method.

Data Set	Number Vertices	Percentage
NICL Baseline Contours	69,712	100.00%
NICL Thin 1.2m	7,134	10.24%
LACA 6K Smooth	22,790	32.69%

Table 1: Number of vertices in original, thinned, and smoothed data sets.

4.1.1 Horizontal accuracy tests method 1: buffer zone tests

The primary test for this method involves a comparison of the horizontal accuracy about the original NICL lines (*NICL Buffers* at 0.6 and 1.2 meters) with the thinned and smoothed results from *LACA 6K Smooth* [1, p.429]. See Tests 1 and 2 in Table A-2.

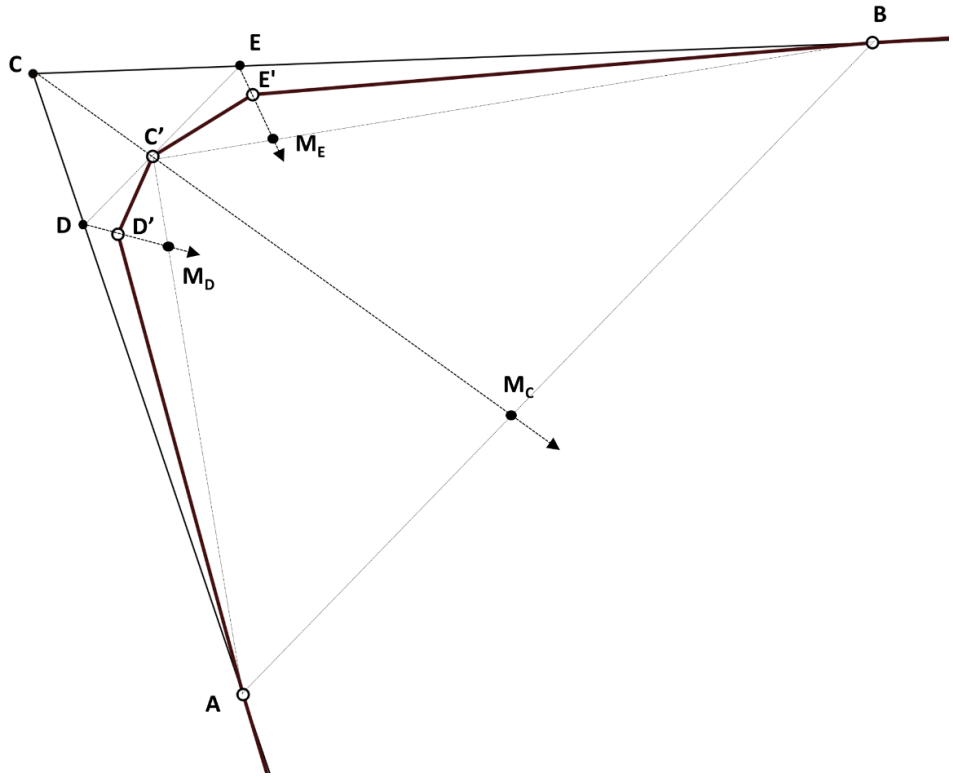


Figure 17: Iterative geometric replication in a selected interval: A vertex C′ was moved along CM_C at a TF of 0.237 in the initial call to the LACA smoothing. Two vertices (D and E) are added to the interval ABC. The new sub-intervals AC′D and BC′E are then recursively evaluated resulting in new smooth points at D′ and E′. The process repeats until the I_T of 0.6 (measured as D′ to M_D and E′ to M_E) is not exceeded.

Specifically, we used the QGIS 3.16 tool "Count Points in Polygon" to count the number of smooth vertices inside the two buffer zones [1, p.978]. The results are shown in Table 2. There were 22,790 vertices in the *LACA 6K Smooth* data set. The overwhelming majority fell within the outer buffer of 1.2 meters from the NICL (22,643) leaving 147 anomalous points that managed to fall outside the bounds of both buffers. This is a 99.36% success rate. However, it falls short of the stated goal of 3 sigma or 99.73% introduced in section 2.5.1 by 0.37% and should be qualified.

In Table 2, the observed percentage of *LACA 6K Smooth* points within 0.6 meters of the line is 88.44%. This comfortably exceeds the expected sigma of 86.64% introduced in Section 2.5.1 meaning that this level of the test for accuracy passed without qualification.

In addition to tests of *LACA 6K Smooth* compared to the *NICL Buffers*, this section also includes the results of accuracy for tests of other smoothing methods discussed here. This includes B-Spline Snakes from a QGIS generalize function (Tests 3 and 4); our implemen-

Test	Reference Buffer	Evaluated Method	Points Inside	Points Outside	Total Points	Pct In	Ranked Sigma
9	*NICL Buffer 0.6 m*	*LACA 50K Smoothing*	139	470	609	0.2282	0.29
10	*NICL Buffer 1.2 m*	*LACA 50K Smoothing*	243	366	609	0.3990	0.52
3	*NICL Buffer 0.6 m*	*B-Spline Snakes Smoothing*	2,924	4,210	7,134	0.4099	0.54
4	*NICL Buffer 1.2 m*	*B-Spline Snakes Smoothing*	4,900	2,234	7,134	0.6869	1.01
5	*NICL Buffer 0.6 m*	*5X Surface Smoothing*	43,129	11,081	54,210	0.7956	1.27
6	*NICL Buffer 1.2 m*	*5X Surface Smoothing*	52,851	1359	54,210	0.9749	2.24
7	*NICL Buffer 0.6 m*	*Kettunen FPDEMS*	51,923	5,323	57,246	0.9070	1.68**
8	*NICL Buffer 1.2 m*	*Kettunen FPDEMS*	56,441	805	57,246	0.9859	2.46
1	*NICL Buffer 0.6 m*	*LACA 6K Smoothing*	20,157	2633	22,790	0.8845	1.58**
2	*NICL Buffer 1.2 m*	*LACA 6K Smoothing*	22,643	147	22,790	0.9935	2.73
	NICL Baseline	—	69,524	188*	69,712	0.9973	3.00

Table 2: Results for horizontal accuracy (Method 1): percentage of smoothed points inside NICL Buffer Zones.
* Allowable number of error points at 3 Sigma.
** Exceeds 1.5 Sigma or 86.64% of points for 0.6-meter buffers.

tation and test of generic smoothing at five iterations (*5X Smooth*—Tests 5 and 6); and, our implementation and test of the Kettunen feature preserving method (*Kettunen FPDEMS*—Tests 7 and 8).

All the tests 1–8 were conducted at a 1:6,000 scale and all were evaluated against the NICL centerlines using the standard 0.6- and 1.2-meter buffers described earlier. None of the methods except *LACA 6K Smooth* come close to the goal of 3.0 sigma for the 1.2-meter buffer, so it seems prudent to rely most on the 0.6-meter buffer tests as a primary means to distinguish our results.

From the results in Table 2, we conclude that any method with a sigma value greater than 1.5 for a buffer of 0.6 meters is significant. Only two methods meet this criterion, the *Kettunen FPDMS* (1.68 sigma) and the *LACA 6K Smooth* (1.58 sigma) methods. While the advantage of the Kettunen result is somewhat surprising, this method does show promise for smoothing as shown in alternative tests in Section 4.2 dealing with contour smoothness.

We must also mention that, as an alternative to considering the 0.6-meter buffer alone, we experimented with averaging the sigma for both 0.6- and 1.2-meter buffers for the *Kettunen FPDEMS* and *LACA 6K Smooth* methods. Considering both buffers, the Kettunen method averaged (1.68 + 2.46) / 2 = 2.070 sigma while the LACA averaged (1.58 + 2.73) / 2 = 2.155 sigma. This is considered an indication that the LACA method excels on average.

Finally, to complete our discussion of the Method 1 tests in Table 2, we must also mention that a test of the LACA method was conducted at a 1:50,000 scale as in Tests 9 and 10 (see *LACA 50K Smooth* in Table A-2). As described in Table A-1, this data set was developed using a thinning tolerance T_T = 10 meters and an insertion threshold I_T = 5 meters, both derived using Equation 1 and the scale of 1:50,000. A contour interval of 5 meters was used for this test, a somewhat tight contour interval for this scale but necessary to show variation over the test area.

As expected, *LACA Smooth 50K* varies considerably from the *NICL Baseline* centerlines, the natural result of a dramatic scale change. As such, it lacks the vertical accuracy originally attributed to the data: generalized results should have an error statement and a qualified $RMSE_h$. In Section 4.2.4, we present 1:50,000 scale test results and suggest a method

to amend the base error statement (i.e., $RMSE_h$ = 0.122) to reflect the impact on accuracy that results from smoothing with scale changes (i.e., a revised $RMSE$ of 0.261).

4.1.2 Accuracy tests method 2: x, y accuracy estimated from vertical deviations

In addition to the buffer zone tests of Method 1 in the previous section, we offer a second method of horizontal accuracy verification described earlier in Section 2.5.3. That is, by developing vertical deviations of the height estimates along the smoothed centerline using the DEM. This test is based on the premise that smoothed points lying close in x and y to the *NICL Baseline* likely agree in height estimated from the underlying DEM owing to the fact that the *NICL Baseline* centerlines precisely match the DEM. By using the DEM to estimate the true elevation at an x and y smoothed point having a nominal contour elevation, we can develop an indirect measure of how closely the lines from a given smoothing method are correlated in x and y to the standard *NICL Baseline* (see the method portrayed in Figure 3).

For this evaluation, we employ hypothesis testing using the Z statistic and probability level for our tests [55]. This is a widely used statistical technique that, in our case, allows us to compute the likelihood that our smoothed data for any given data set adheres closely enough to the *NICL Baseline* data.

Hypothesis testing for two groups is based on a null and an alternative hypothesis [55]. The null hypothesis for this test is that the mean deviations in heights for a tested smoothing method are not significantly different from the mean deviations of the *NICL Baseline*. If we cannot reject the null hypothesis, then we may conclude that the lines are indistinguishable in terms of proximity to one another.

Formally stated the null hypothesis is: $H0 : \mu_{SMOOTH} = \mu_{NICL}$.

The alternative hypothesis is a two-tailed test stating that we expect the mean of the smoothed data to be either significantly higher or lower than that of the NICL Baseline data. We set a standard for the probability (P) that we have reached the right conclusion with regard to the null hypothesis at 0.05. $HA : \mu_{SMOOTH} \neq \mu_{NICL}$ at P = 0.05.

If we reject the null hypothesis, the mean deviation for the smoothing method in question is different enough from that of the NICL Baseline to conclude that the smoothed lines violate the accuracy of the original data. Refer to Table 3.

Smoothing Method	Average Deviation	SD	N	Z-Test Score	P-Value
LACA 50K Smooth	0.04423	0.261	609	4.1817	0.0000289*
B-Spline Snakes	0.01291	0.176	7,134	6.0532	0.0000000134*
5X Smoothing	-0.01000	0.470	54,210	-4.8262	0.0023347*
Kettunen FPDEMS	-0.00229	0.340	57,246	-1.5346	0.1248716
LACA 6K Smooth	0.00017	0.235	22,790	0.1042	0.9169800
NICL Baseline	0.00000	0.122	69,712	—	P = 0.05

Table 3: Results for horizontal accuracy (Method 2): test of significance from height deviation estimates.
* Reject the Null hypothesis: the lines differ significantly from NICL Baseline in x and y.

While the range for average deviations in Table 3 appears small, the values differ by orders of magnitude. The test involves consideration of the number of vertices in each

group and the standard deviations of the groups as well. The relative ratio of these values determines a Z-Test score as follows:

$$ZTest = \frac{Avg_{NICL} - Avg_{SMOOTH}}{\sqrt{\frac{{SD_{NICL}}^2}{N_{NICL}} + \frac{{SD_{SMOOTH}}^2}{N_{SMOOTH}}}} \tag{13}$$

where

- Avg_{NICL} = average height deviation of the NICL Baseline (zero for the baseline—the NICL values match the DEM perfectly)
- Avg_{SMOOTH} = average height deviation for the smoothing method in question
- SD_{NICL} = standard deviation of the original NICL angles
- SD_{SMOOTH} = standard deviation of the smoothing method in question
- N_{NICL} = number of points in the original NICL data
- N_{SMOOTH} = number of points in the smoothing method in question

From Table 3, we reject the null hypothesis that the means are equal for the first three methods. We conclude that the indicated methods have introduced significant changes to the x and y location of the smoothed lines based on the inference from height displacement. Conversely, we accept the null hypothesis for the *Kettunen FPDEMS* method and the *LACA 6K Smooth* method meaning that these two are equivalent to the *NICL Baseline* results: they are close enough to be matched to the *NICL Baseline* in x and y. Note that the P-value for *LACA 6K Smooth* is particularly high in relation to the other methods.

Finally, we present graphic evidence supporting Table 3 results in Figure 18. The order of the ranked results in the figure supports the numbers from the Table. Also particularly noteworthy, the *LACA 6K Smooth* results consistently adhere to the turnbacks in the line. *Kettunen FPDEMS* remains close to the *LACA 6K Smooth* method, while *5X Smoothing* and, to a greater extent, *B-Spline Snakes*, retain only remnants of the turnback shapes.

4.2 Tests of contour smoothness

Having evaluated the horizontal accuracy of the smoothed lines and identified *Kettunen FPDEMS* and *LACA 6K Smooth* as those methods that retain accuracy, we now turn our attention to the smoothness of the lines. For completeness, we again test all smoothing methods, even those that failed the accuracy tests. Specifically, we will test whether or not the smoothing methods have reduced the angularity and thus improved the smoothness of the lines.

In Section 2.5, we described proposed tests of the enclosed angle which we will use as a basic metric of line angularity. From these metrics, we develop basic statistics including the mean angle and the standard deviation of the angles of the lines. The statistics from these measures can then be used in hypothesis testing similar to the methods we employed in the previous section on x and y accuracy testing.

We illustrate the tests on smoothing by walking through the comparison of the *LACA 6K Smooth* results to that of the *NICL Baseline*. In our case, hypothesis testing allows us to compute the likelihood that our "smoothing treatment" of the thinned data produced smoother contours than can be found in the original NICL data. Hypothesis testing for two groups is again based on a null and an alternative hypothesis. The null hypothesis is that the angularity is not significantly different between the two datasets and that the

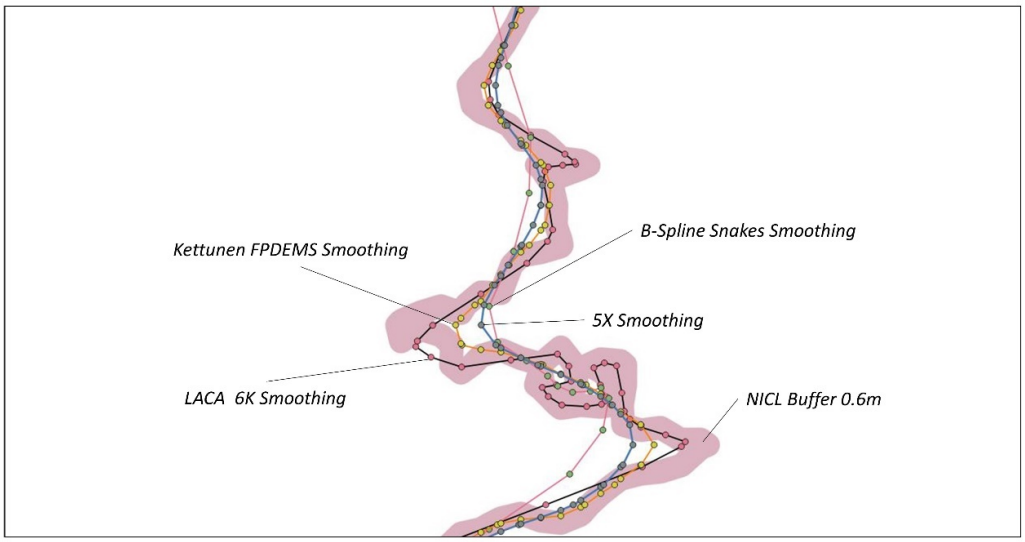


Figure 18: Composite relationships of smoothing methods at 1:6,000 scale. A representative sample of the relation between various smoothing methods is examined here with the *NICL Buffer 0.6m* as a backdrop. Notice the order of the lines, particularly as they fan out at turnbacks. This is a consistent pattern throughout the area: *LACA 6K* lies closest to the buffer centerline followed by *Kettunen FPDEMS* then *5X Smoothing* and *B-Spline Snakes*.

mean angle for the *LACA 6K Smooth* data is not significantly different from that of the *NICL Baseline* after our smoothing.

Formally stated, the null hypothesis for Test 16 is: $H0 : \mu_{LACA} = \mu_{NICL}$.

If we cannot reject the Null hypothesis, then we must conclude that the lines are indistinguishable in terms of angularity; that we have not significantly reduced angularity and thus improved smoothness using our method.

4.2.1 Initial smoothing tests for *LACA 6K Smooth* versus *NICL Baseline*

In Table 4, we present our analysis of the degree to which the *LACA 6K Smooth* contours have reduced the angularity of the *NICL Baseline* contours, that the LACA contours are significantly less angular than the NICL data. In this Table, we are comparing the original NICL (less the insignificant small polygons), which is composed of densely threaded contour vertices, to the thinned and then reconstituted smooth LACA lines. Again, the basic criterion for the comparison is the result of computing the enclosed angle of each internal vertex in each three-point sequence in the lines (refer to Section 2.5 and Figure 4). These comparisons form the individual observations that are cumulatively tested for significance here.

The alternative hypothesis is a unidirectional (one-tailed) test stating that we expect the mean of the LACA angles to be closer to 180 degrees than that of the NICL data. $HA : \mu_{LACA} > \mu_{NICL}$ at P = 0.05.

Statistic	NICL Baseline	LACA 6K Smooth
Average	179.7697	179.9109
Min	0.0000	8.8160
Max	358.1473	359.3293
SD	20.1579	25.8634
N	69712	22698
Kurtosis	14.7578	3.8037
Skewness	-0.2358	0.0243
Z-Test	-0.7516	
P-Level	0.2290*	
Sig Level	0.05	

Table 4: Test 16: test of mean angles—*NICL Baseline* versus *LACA 6K Smooth*.
* Since P-Level exceeds Sig Level, we cannot reject the null hypothesis meaning there is essentially no difference between the means of the two datasets. Both datasets are equally smooth.

Specifically, we expect that our smoothing has had an effect and that the mean value for the smoothed LACA angles is closer to 180 degrees than the NICL value. As before, the test involves consideration of the number of vertices in each group and the standard deviations of the groups as well. We determine a Z-Test score as before and specifically as follows:

$$ZTest = \frac{Avg_{NICL} - Avg_{LACA}}{\sqrt{\frac{{SD_{NICL}}^2}{N_{NICL}} + \frac{{SD_{LACA}}^2}{N_{LACA}}}} \tag{14}$$

where

- Avg_{NICL} = average of the original *NICL Baseline* angles (no thinning applied)
- Avg_{LACA} = average of the *LACA 6K Smooth* angles
- SD_{NICL} = standard deviation of the *NICL Baseline* angles
- SD_{LACA} = standard deviation of the *LACA 6K Smooth* angles
- N_{NICL} = number of enclosed angles in the *NICL Baseline* data
- N_{LACA} = number of enclosed angles in the *LACA 6K Smooth* data

The Z-Test value of -0.7516 from Table 4 renders a probability level of 0.2290 which is greater than the significance level chosen of 0.05. From Test 16, we must conclude *that there is no significant difference between the two means*, and we are unable to prove that the *LACA 6K Smooth* has meaningfully reduced angularity in the original NICL data.

4.2.2 Test of *LACA 6K Smooth* Versus *NICL Thin 0.5m* data

Upon closer examination of the distribution of the *NICL Baseline* data, however, the extreme leptokurtic nature of the NICL data stood out. By a ratio of 4:1, angles in two bins ranging from 177.5 to 182.5 degrees occurred more frequently than all other adjacent bins (roughly a combined 31,000 of 69,000 observations– see Figure 5b in Section 2.5).

The explanation we offer for this finding is that the original NICL Baseline data are oversampled. The threading process described earlier interpolates a point at each neighborhood edge crossing regardless of whether or not an actual inflection in the angle of the

line exists. Normally, this is a length of less than one meter in x and y spacing between vertices. Many of the original points interpolated along neighborhood edges are completely colinear and thus redundant. The result is an over-representation of angles near 180 degrees (see Figure 5b in Section 2.5).

To test this theory, we applied minimal thinning to the original NICL data and repeated Test 16 as Test 17. The essential question to be answered before committing to this approach was what value of T_T should we use to perform this thinning? For general guidance, the value selected for thinning should be less than the insertion tolerances used to create the *LACA 6K Smooth* data: we do not want to risk undersampling the *NICL Baseline* data. The value 0.6 meters was used for the insertion of a smoothed LACA point. This suggested something slightly less than 0.6 meters, so a minimal value of 0.5 meters was selected and applied to the *NICL Baseline* data rendering a new thinned dataset *NICL Thin 0.5m*.

A comparative view of the *NICL Thin 0.5m* and *NICL Baseline* vertices is shown in Figure 19. We concluded that the *NICL Thin 0.5m* is a truer picture of the angularity in the data that was masked in the oversampled, unthinned *NICL Baseline* data. We also concluded that a new test, Test 17, was in order using the *NICL Thin 0.5m* data. See the results for Test 17 in Table 5.

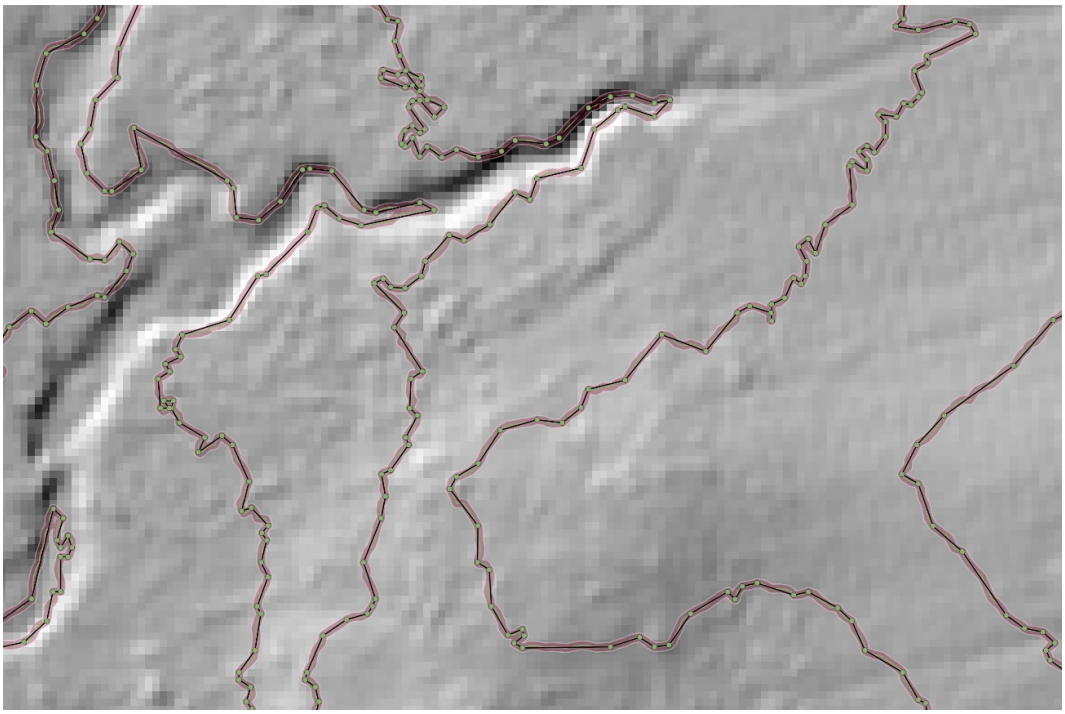


Figure 19: *NICL Thin 0.5m*: deriving an estimate of underlying angularity. The thin black line and retained green vertices are the *NICL Baseline* after 0.5m thinning (*NICL Thin 0.5m*). The red band is the 0.6m buffer. The scale of the display is 1:600.

In the re-test shown in Table 5, the Z-test score of -2.2458 yields a P-value of 0.0247, a value less than the significance level of P = 0.05. In this case, we can reject the null hypothesis that the means are equivalent. We conclude that *there is a significant difference between the*

Statistic	NICL Thin 0.5m	LACA 6K Smooth
Average	179.1676	179.9109
Min	0.2923	8.8160
Max	359.2856	359.3293
SD	37.9049	25.8634
N	17942	22698
Kurtosis	2.2858	3.8037
Skewness	-0.0845	0.0243
Z-Test	-2.2458	
P-Level	0.0247*	
Sig Level	0.05	

Table 5: Test of mean angles: Test 17—*NICL Thin 0.5m* and *LACA 6K Smooth.*
* P-value is significant at the 0.05 level.

two means, specifically that *LACA 6K Smooth* data has a meaningfully larger average angle once the initial extreme density of the original line is considered. By judiciously thinning colinear vertices, we have removed many of the spurious angles occurring very near 180 degrees evident in the original *NICL Baseline* data.

More specifically, the mean value of the *LACA 6K Smooth* data is closer to 180 degrees and exhibits a much smaller variation about the mean. Again, we interpret this as an indication that the LACA smoothing has reduced the angularity inherent in the original NICL data and has generated smoother lines.

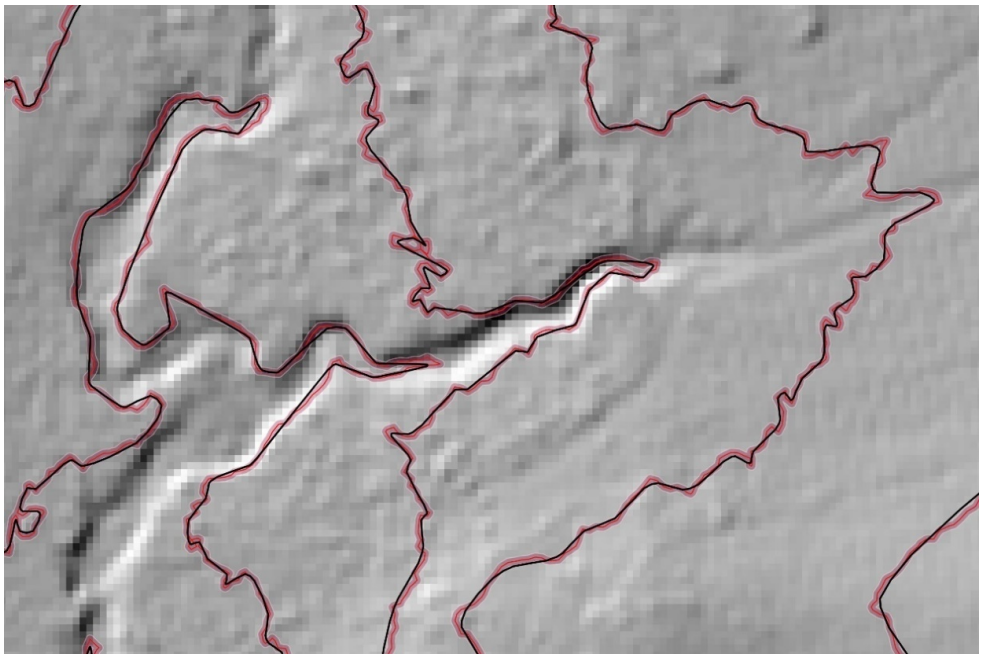
The table also supports the contention that, for a comparatively modest increase between the number of points in the thinned NICL data and the smoothed data, there is a sizable increase in the quality of the displays (see Table 1). The smoothing about the medial axis seems to insert the right points at the right place producing a better result. Compare Figures 12 and 13 and Figures 20a and 20b.

4.2.3 Tests of smoothness for other smoothing methods

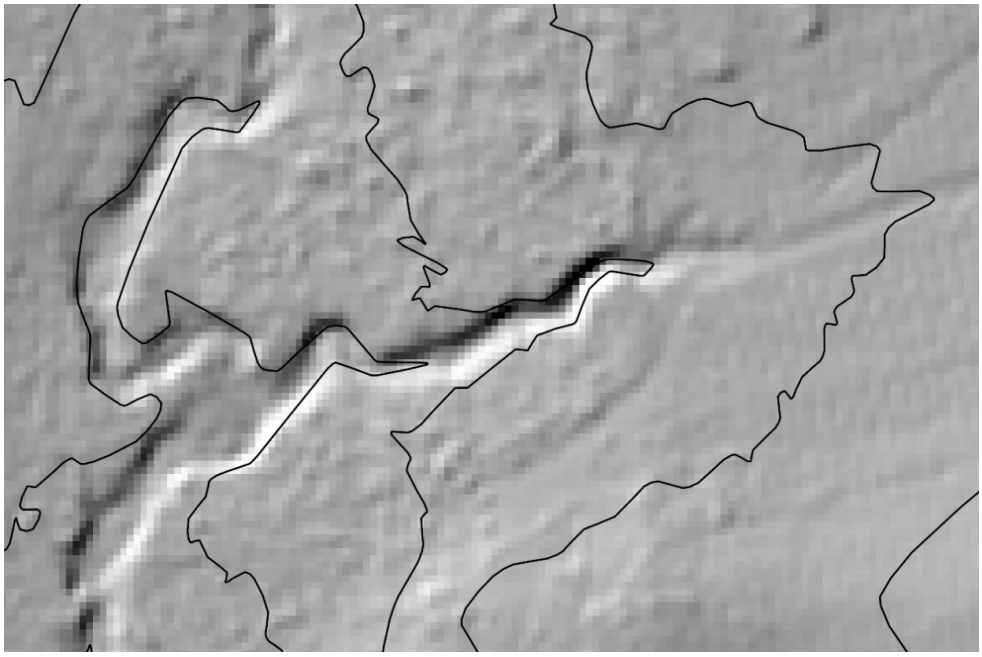
Smoothing tests were also applied to the other methods of smoothing. There were two sets of tests performed with results from set one shown in Table 6 and the second set shown in Table 7.

The first set (Tests 18, 20, and 22 in Table A-2 in Appendix A) compared each of the smoothing methods *B-Spline Snakes*, *5X Smoothing*, and *Kettunen FPDEMS* to the same data used as a reference for the *LACA 6K Smooth* in Test 17. That is, the tests were intended to determine whether or not the smoothing methods showed less angularity and were thus smoother than the reference data in *NICL Thin 0.5m*. We believe this latter data set to be representative of the true degree of angularity in the NICL Baseline From Table 6, all the methods produce smoother results than the *NICL Thin 0.5m* dataset.

The second set of tests (19, 21, and 23) were conducted to see if there were differences in the degree of smoothness between the three alternative smoothing methods listed above and the *LACA 6K Smooth* method. We want to be able to answer the question of whether or not the smoothness produced by the *LACA 6K Smooth* method is as smooth as any of the other methods. As a result, the *LACA 6K Smooth* is used as the reference data set. We found no difference in the degree of smoothness for any of the three methods.



(a) *NICL Baseline* with *LACA 6K Smoothed*.



(b) *LACA 6K Smooth* contours alone.

Figure 20: The scale of the display is 1:600.

Test #	**18**	**20**	**22**	**Reference Dataset**
Statistic	B-Spline Snakes	5X Smoothing	Kettunen FPDEMS	NICL Thin 0.5m
Average	179.9583	180.0668	180.2606	**179.1676**
Min	0	54.9808	13.3959	**0.2923**
Max	345.2714	324.1131	355.0916	**359.2856**
SD	18.6127	7.7315	12.3048	**37.9049**
N	6950	53945	56930	**17942**
Z-Test	2.1936	3.1559	3.7963	
P-Value	0.02826*	0.0016*	0.000147*	
Sig-Level	0.05	0.05	0.05	

Table 6: Tests of smoothness for three methods using *NICL Thin 0.5m* as reference.
* P-Value is significant at the 0.05 level. In each test, we can reject the null hypothesis that there is no difference in the angularity between each successive smooth test and the reference data set. All the tested methods produce smoother results than the *NICL Thin 0.5m* reference data.

Test #	**19**	**21**	**23**	**Reference Dataset**
Statistic	B-Spline Snakes	5X Smoothing	Kettunen FPDEMS	LACA 6K Smooth
Average	179.9583	180.0668	180.2606	**179.9109**
Min	0	54.9808	13.3959	**8.8160**
Max	345.2714	324.1131	355.0916	**359.3293**
SD	18.6127	7.7315	12.3048	**25.8634**
N	6950	53945	56930	**22698**
Z-Test	0.1682	0.8914	1.9504	
P-Value	0.8664	0.3727	0.0511	
Sig-Level	0.05	0.05	0.05	

Table 7: Tests of smoothness for three methods using *LACA 6K Smooth* as reference.
The P-Value is not significant at the 0.05 level for any test. In each test, we must accept the Null hypothesis that there is no difference in the angularity between each successive smooth test and the reference LACA 6K Smooth data set. Note, however, that *Kettunen FPDEMS* came very close to being smoother than *LACA 6K Smooth* in Test 23.

We had expected that either the B-Spline Snakes or the *5X Smoothing* may be significantly different from the *LACA 6K Smooth* dataset indicating that these datasets were overly-smoothed. We did not find any such indication.

4.2.4 An experiment with LACA Smoothing at 1:50,000 scale

The proposed method is subject to concerns that it can cause crossing contours or that it will cause an unacceptable number of coalesced contours at significantly reduced scales. To test these possibilities, an additional smoothing experiment was conducted using an interval of 5 meters and at a 1:50,000 scale. To review, the tolerance and thresholds for thinning and insertion must be recalculated when scale changes are needed. From Equation 1, the thinning tolerance and insertion threshold for the 1:50,000 scale are computed as

$$T_T = 50,000 * 0.2/1000 = 10000/1000 = 10 meters$$

and

$$I_T = T_T * 0.5 = 5meters$$

4.2.5 An amended accuracy statement for 1:50,000 scale results

After each change to the tolerance for minimal thinning, only the significant points determined by thinning and smoothing remain. We feel that a scale change such as this necessitates a revision of our accuracy statement: smoothing and displacing the lines in x and y has implications for the implied height accuracy of the data. Even generalized contours have an accuracy requirement.

As an amended statement of $RMSE_h$, we propose the use of the methods described in Section 4.1.2. There, we estimated the vertical displacement of the relocated smoothed points by projecting the x and y of the smoothed points onto the original DEM. These variances in height are a revised statement of vertical accuracy in the data.

From Table 3, the standard deviation for the *LACA 50K Smooth* result is 0.261 meters. Compared to the original DEM $RMSE_h$ of 0.122, this is a substantial increase. The 95% confidence interval for the smooth data is 1.96 * 0.261 = 0.512 meters. Considering the magnitude of scale change, however, this is actually a benign result when compared to the accuracy of the other methods (see Table 3). The LACA adherence of the smoothed line to the original pays dividends when affecting a scale change.

In general, however, the LACA method fails to produce satisfactory results for the 1:50,000 scale display. While the method works well over most of the test area, it fails in at least two prominent, elongated turnbacks shown in areas A and B in Figures 21 and 22. The muted orange buffer in Figure 22 is 10 meters in width on the ground (i.e., 0.2 mm or a line thickness at 1:50,000 scale). The overlapping, self-intersecting buffer lines represent self-coalescing sections of the lines. A cartographer would likely further **simplify** the problem areas by snipping the lines along segments **ab**, **cd**, and **ef** in Figure 22.

The presence of objectionable coalesced sections in the 1:50,000-scale experiment requires a closer examination. For details, see the proposed algorithm and data structures in Appendices B and C.

4.3 An experiment with a one-half meter contour interval

At the 1-meter interval used in our *LACA 6K Smooth* tests, no crossing contours were observed. However, we know that the proposed method as implemented can potentially create such occurrences. In order to stress-test the possibility of crossings, we experimented with a denser contour interval of 0.5 meter. The $RMSE_h$ accuracy of our survey data supports such an interval per the guidance of Flood [2].

$$SCI = AccuracyH/0.5958 \tag{15}$$

where SCI is Supportable Contour Interval and $AccuracyH = 1.96 * RMSE_h$.

Using the error values from the study: $SCI = 1.96 * RMSEh/0.5958 = 1.96 * 0.122/0.5958 =$ minimum 0.4020-meter interval. Since 0.5 > 0.4020, we are safe to use a contour interval of 0.5 meters for this test.

The results of this experiment are shown in Figure 23 which closely examines areas of densely spaced contours in the test area. No crossing contours were observed. Note that our method still has the liability to cause crossing contours given the right combination of

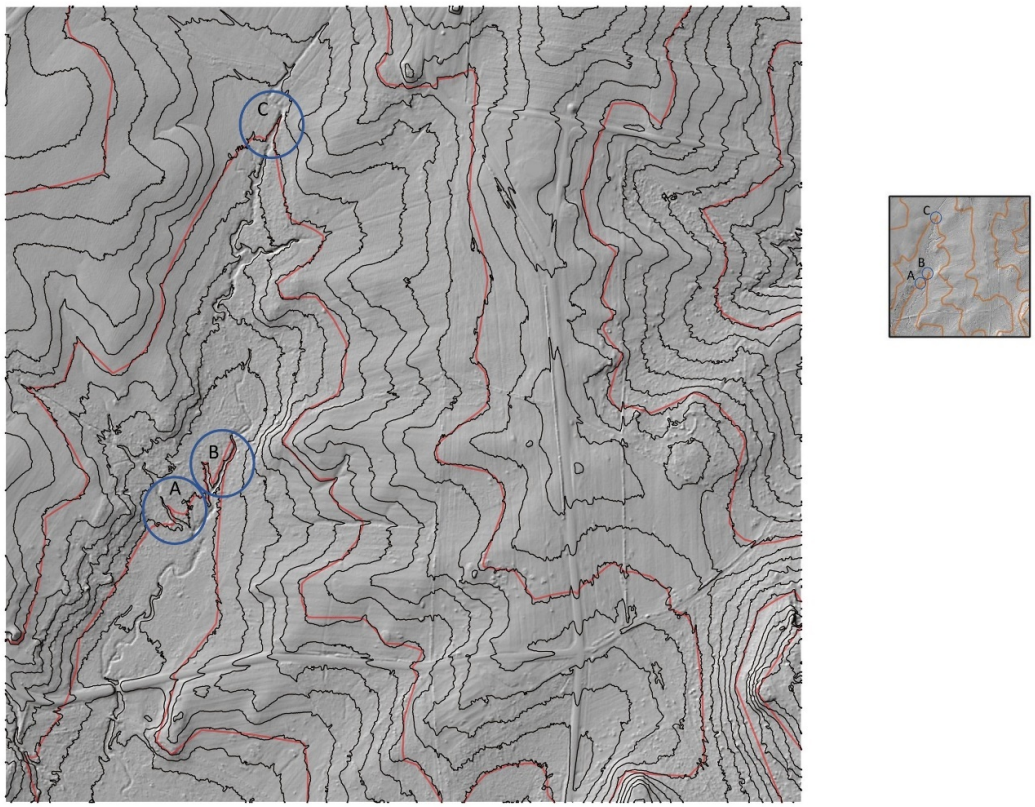


Figure 21: Comparison of the *LACA 50K Smooth* to the *NICL Baseline*. The red line represents the 5-meter contour lines for the 1:50,000 scale display (*LACA 50K Smooth*). The black lines represent the *NICL Baseline* at a 1-meter interval. The scale of the main display is 1:6,000. The blue circles represent areas of possible coalescence and the smaller figure is the image at 1:50,000 scale. In the smaller image, the lines are coalesced, particularly the narrow turnbacks at A and B.

unfortunate geometric circumstances, but our dynamic calculation of a smaller smoothing displacement in areas of steeper slopes ameliorates this likelihood.

4.4 Notes on smoothed contour line symmetry

Section 2.5.5 described the proposed methods of testing our *LACA Smooth 6K* results for symmetry: the state of having matched vertices falling equally about the medial axis of an interval in the smoothed contour line. While contour lines are arguably irregular by definition (we are dealing with *symmetry* of *asymmetric* shapes), the method of testing described in section 2.5.5 proposed normalized measures of symmetry. That is, we use the *relative percentage* of angles or *relative percentage* of projected distances along the original lines to assess symmetry using regression analysis (see Figures 6 and 7).

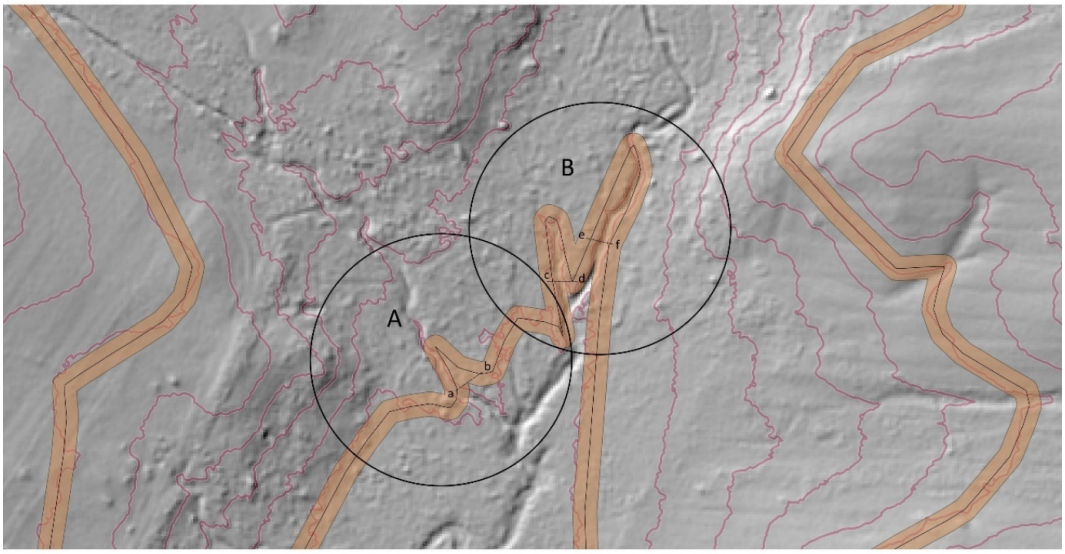


Figure 22: Enlarged view of coalesced sections of a 1:50,000 scale line. The detailed purple lines represent the *NICL Baseline*, the thin black line represents the *LACA 50K Smooth* reconstituted using an I_T = 5 meters, and the light orange buffer represents the ground distance of the smoothed line thickness at 1:50,000 scale.

The accepted method of evaluating the significance of a regression analysis is hypothesis testing using a t-test statistic (see the test description in Table A-3). The null hypothesis for the three regression tests described here assumes that there is no linear relation between the two variables being compared. That is, that the slope of the line approximating points in a scattergram as seen in Figures 24–26 equals 0 and is a horizontal line drawn through the scattered points at a constant mean y-value.

The alternative to the null hypothesis is that a linear relationhip exists between the variables. More exactly for our case, we expect that the slope of the line approaches 1.0 or that the dependent variable y (i.e., the *next angle* equals the independent variable or x (i.e., *previous angle*) in our scattergram. A perfectly matched set of observations of corresponding angles or projected distances for vertices in a line interval can be plotted on an x and y axis as a straight line with the equation $y = x$ (see the dashed red line in Figures 24). This is the *identity line* and is considered to be a condition of perfect symmetry for corresponding points on either side of the angle bisector.

In Figure 24, the observed relationship between previous and next angles is given by y = 0.5981x + 0.201. This is the *regression line*. The slope of the line is positive: as the previous angle increases, there is a moderate increase in the corresponding next angle. However, it is far from a perfect relationship. The slope of the line is 0.5981 compared to 1.0, the slope of the identity line. The regression line, even after setting the y- intercept to 0 and forcing the line to match at the onset with the identity line (see the equation $y = 0.9273x$ in Figure 24), deviates from the ideal. The *strength* of the relationship is given by the R^2 term (the *coefficient of determination* or the amount of variation explained by the previous

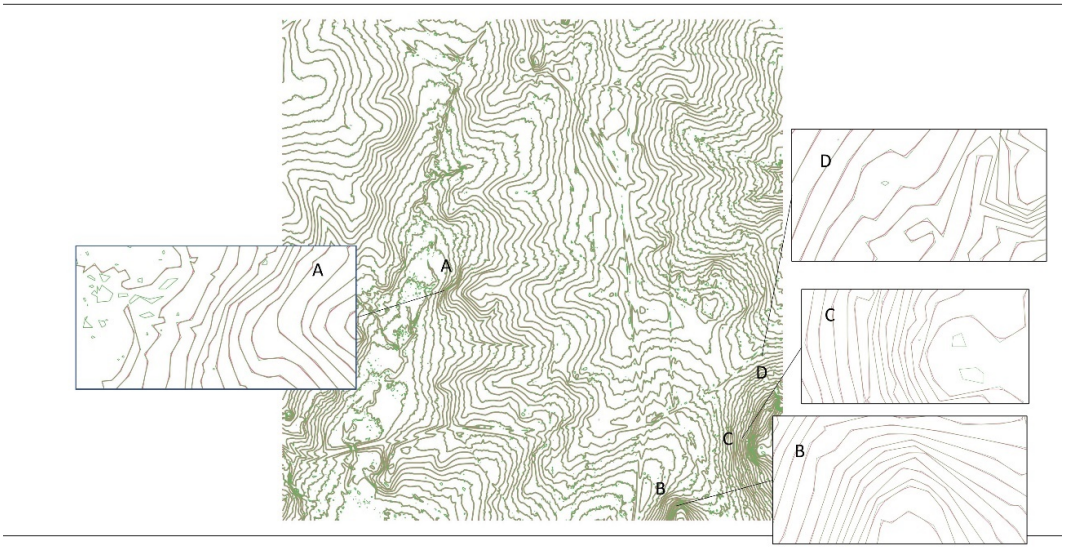


Figure 23: *LACA 6K Smooth* contours at one-half meter contour interval. A 0.5-meter contour interval in steep areas shows no crossing contours despite the high potential for such occurrences.

angle as a predictor of next angle) is a modest 0.3598. The coefficient of determination must always be evaluated in an application context and here we are dealing in inexactitude with a phenomenon that is the epitome of irregularity. In spite of the vagaries of the line geometry, however, we expected a better fit.

Despite the noted weaknesses of the coefficent of determination, a statistical test of significance of the relation revealed that it is different from 0; that a weak linear relationship exists (see Table 8). The slope of the line (0.5981) is much different from the ideal value of 1.0 and thus analysis of symmetry by angles is considered a weak relation and we look to an alternative test method for better results.

Statistic	Value
$\mathbf{R^2}$	0.3598
n	4216
Slope	0.5981
Y-Intercept	0.2010
t-value	48.6659
p-value	0*

Table 8: Test of significance for relative angles.
* The p-value is 0 to 8 significant digits and is less than the significance level of 0.05. We can reject the null hypothesis: a weak linear relation exists.

The second method of symmetry analysis is shown in Figure 25. It was obtained by examining the relationship of previous to next relative distance by projecting the smoothed

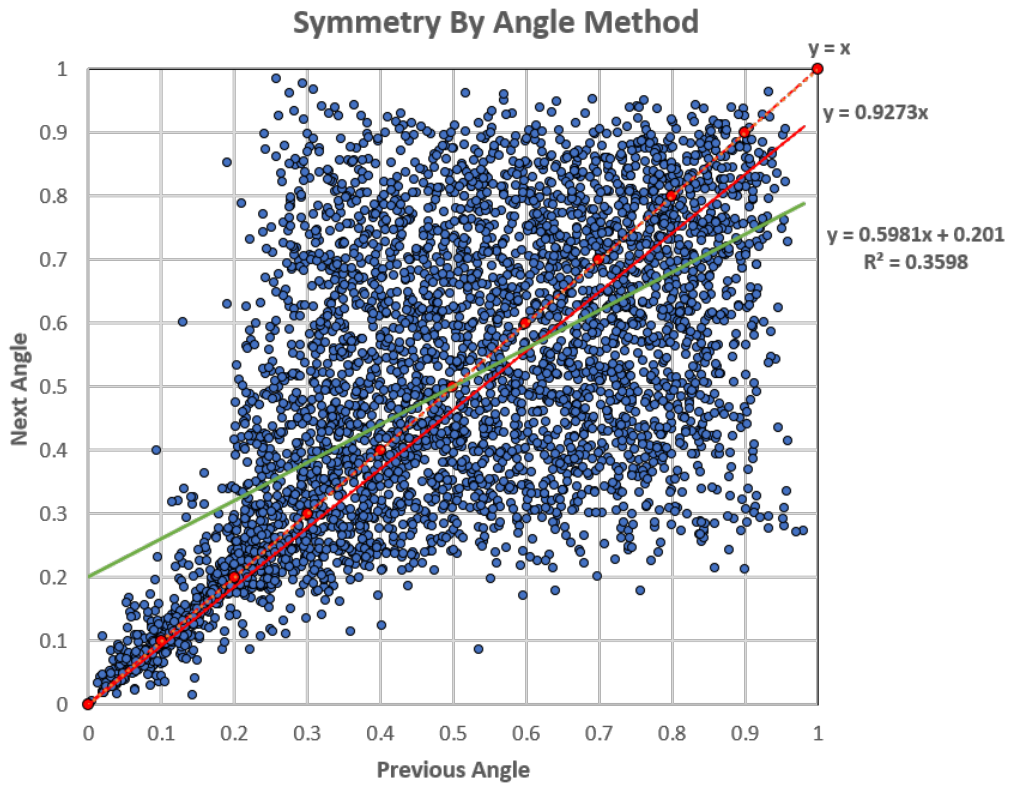


Figure 24: Symmetry measured by the angle method.

points from *LACA Smooth 6K* onto the original *NICL Thin 1.2m* bounding lines. See the explanaton of the concept of relative position and projected distance in Figure **??**. As with angles, the intervals of the line provide the basic unit of analysis and the projected distance percentage falls with the range of 0 to 1. In general, this method offers more encouragement than provided by the assessment of angles.

The coefficent of determination for this method is a fairly strong 0.7858. However, as shown in Figure 25, there is an unusual concentration of points in the scattergram near 0.4. Some 42% of the original points fall within the range 0.4 to 0.6.. The value of 0.4 is the precise value used for a default tension factor, TF, and this concentration of points undoubtedly artificially boosts the coefficient of determination. To avoid this problem, and to develop what we believe to be a truer picture of the relationship, the limits for analysis were set at 0 to 0.4 excluding the points in the identity range of 0.4 and above (see Figure 26). Here, the coefficient of determination is reduced to 0.7607, still a moderately strong relation and one not affected by the concentration of points at 0.4.

The distribution of points in Figure 26 still show a spread of values, but, at its core, the graphics for the equations developed for this relation show its strength. The identity line (dashed red line in the figure) closely matches the distinguishable linear point distribution

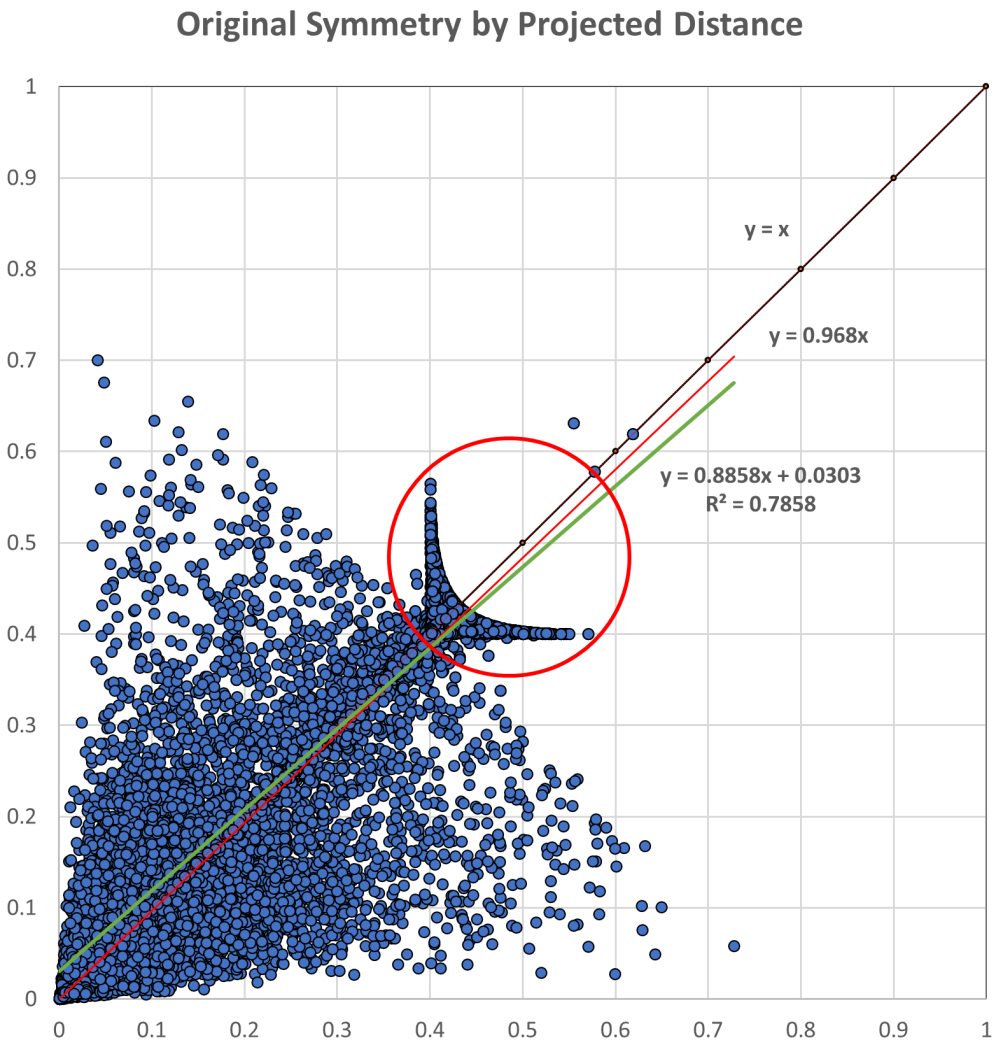


Figure 25: Symmetry measured by projected normalized distance. Note the unusual distribution of points concentrated at 0.4 to 0.5 along both axes. The value for the tension factor, TF, defaults to a value of 0.4 in relatively flat terrain. The projected distance for points in this range is roughly the same yielding the concentration of points near 0.4 to 0.5, particularly strong near 0.40 to 0.43. This biases our result.

evident in Figure 26. The slope of the line (0.8692) is closer to 1, the ideal condition of symmetry.

A statistical analysis of significance of the relation indicates that it is reasonably strong (see Table 9). The t-test value (the appropriate test for regression analysis) is effectively zero: the p-value is less than the alpha level of 0.05 meaning that can reject the null hypothesis that no linear relation exists between our variables. The relative distance of smoothed

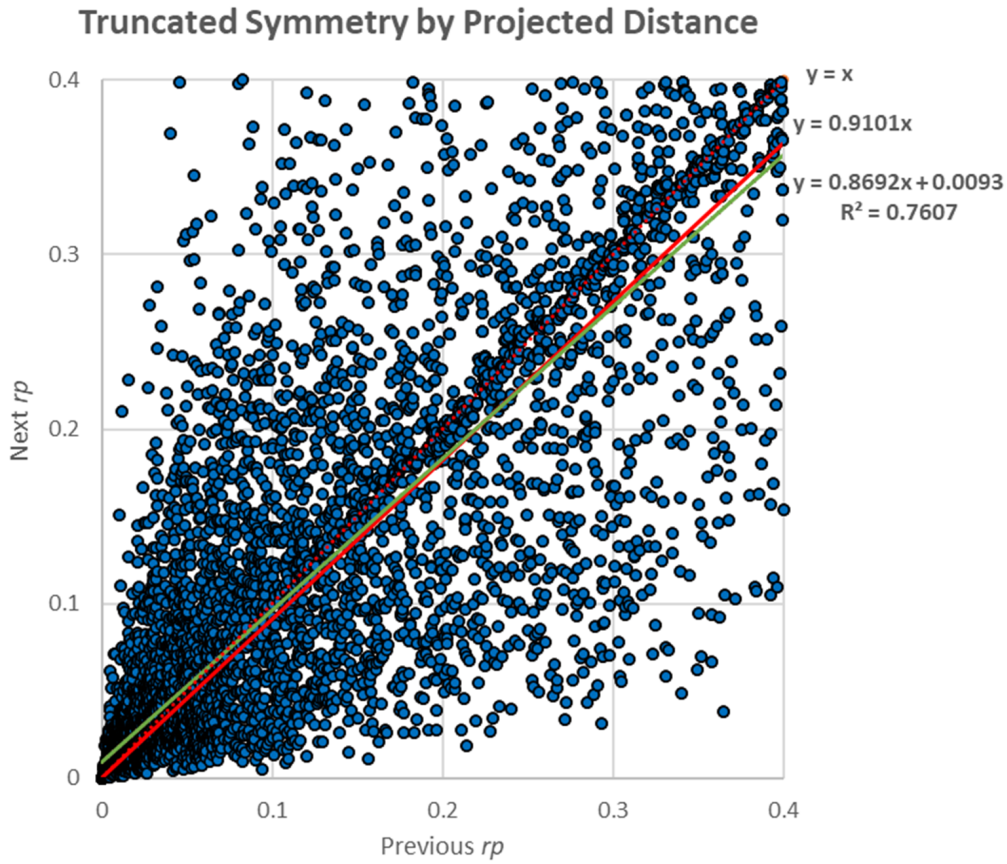


Figure 26: Truncated symmetry measured by projected normalized distance. The dotted red line is the identity line and a tighter clustering of points is evident about this line. The solid red line is offered for comparison only and represents the regression line with y-intercept forced to 0, 0. The green line is the actual regression line which has a y-intercept very close to 0, 0 and a slope approaching 1. This is a strong relation.

points projected onto the original thinned vertices supports the notion that there is symmetry in the intervals and cumulatively in the lines.

5 Summary and conclusions

We believe the method presented here offers a viable alternative to existing methods of achieving smooth contour results, particularly for larger scale applications. We have presented a method that is *smoother* than the source contours, more *accurate* than other alternative smoothing methods, and is defensibly *symmetrical* This research has presented a methodology that is consistent with the precepts of digital cartographic generalization enumerated by Weibel and Dutton [62] and McMaster and Shea [38]. Given the performance of the method, it can be performed interactively and thus meets the criteria for "on-the-fly

Statistic	Value
$\mathbf{R^2}$	0.7607
n	4617
Slope	0.8692
Y-Intercept	0.0093
t-value	121.11
p-value	0*

Table 9: Test for significance of relative position along a line.
* The p-value is 0 to 8 significant digits and is less than the significance level of 0.05. We can reject the null hypothesis: a moderately strong linear relation exists.

generalization" described by Weibel and Burghardt [61]. Specifically, it addressed what is recognized as the first two steps in generalization, *simplification* and *smoothing*.

It does not directly resolve the issue of crossing or self-coalescing sections of a line. The resolution of such problems requires more advanced structures and qualifies as more refined forms of generalization. As a start in this direction, a data structure capable of supporting more refined methods of generalization has been presented in Appendix B along with a basic algorithm for clipping coalesced sections of the lines using contours enmeshed in a TIN (Appendix C).

In Section 4 of the paper, we demonstrated through our results for the *LACA Smooth 6K* that we have met the dual criteria of balancing smoothness and accuracy at the larger test scale of 1:6,000. Testing the method at a reduced scale of 1:50,000, however, showed errors of self-coalescence that can best be addressed by adapting the LACA method to a TIN model based on the contour line segments (see Figures B-2 to B-5 in Appendix B).

We also established through our tests that the methods of smoothing examined here can be ranked in terms of their accuracy and appeal. The *LACA 6K Smooth* results ranked first closely followed by the *Kettunen FPDEMS*. We must acknowledge that feature-preserving surface smoothing approaches have merit, though they too lose adherence to the original, most accurate line at turnbacks (see Figure 18). The *5X Smoothing* and *B-Spline Snakes* show a definite loss of accuracy and significant loss at turnbacks.

There remains further work, however, including the following.

1. We need additional testing on terrain other than the gentle, rolling hills we have tested in this prototype. For example, testing in steeper terrain within the study area from the Gettysburg database with a one-meter contour interval should test any predilection of the method to introduce crossing contours and our use of the local slope to avoid such problems.
2. We conducted this investigation with an accurate and dense LiDAR DEM, but the methods used here could be applied to any other DEM regardless of their source. This work needs to be repeated for other types of DEMs such as the National Elevation Dataset (NED) at 3- and 10-meter post spacing with scale reductions of 1:10,000 and 1:24,000 and vertical errors of 1.82 and 3.04 meters, respectively [53].
3. Finally, in Appendices B and C, we touched on the potentially benign impacts of embedding the contours into a topological TIN model to provide a neighborhood context for operations and to avoid introducing thinning or smoothing errors and self-coalesced sections of the smoothed lines. We demonstrated how such a model

might be used in section Appendix B and in pseudocode illustrating an algorithm to remove self-coalesced sections of lines in Appendix C. Further development and implementation of this model is needed.

Regarding this last point, we suspect that this underlying representation could be useful in many GIS applications and would support a "*contour terrain model*" (i.e., a CTM—contours enmeshed in a TIN model) that could become a third alternative to either DEMs or conventional TINs alone for terrain modeling in the context of a GIS. The CTM would be a good vehicle for interactive displays of topography at working scales in support of analysis.

Given the affinity between contour representations and human perception, this last observation seems a natural evolutionary step in terrain modeling. It is clear, however, that this model would require at least two modifications if it is to be successful.

First, we have to intentionally violate the coveted principle of Delaunay triangulation if the model is to be true to the land surface. Some manipulation of existing triangulation methods is in order, extending even beyond the normal "constrained" model which honors feature edges in the triangulation, but does not remove "flat" triangles that occur when all three vertices of a candidate triangle are from the same contour line or lines of equal elevation [58]. Importantly, the CTM must include conventions for ordering the lines, vertices and TIN edges, something missing from TIN models today.

Second, this will also have to include the derivation and inclusion of geomorphic features such as drains much like the approach of Gokgoz and Gulgen [17] where the contours formed the basis for deriving drainage information. This modification is necessary to remove flat triangles and improve the fidelity of the terrain model to real landforms in the surface.

These ideas need to be explored further, both to avoid possible errors in contour smoothing and for the general utility of a CTM model to GIS applications.

References

[1] QGIS Desktop 3.16 Users Guide, 2022. https://docs.qgis.org/3.16/pdf/en/QGIS-3.16-DesktopUserGuide-en.pdf.

[2] AMERICAN SOCIETY FOR PHOTOGRAMMETRY AND REMOTE SENSING. ASPRS positional accuracy standards for digital geospatial data. *Photogrammetric Engineering & Remote Sensing 81*, 3 (2015), 1–26.

[3] BERTAMINI, M., PALUMBO, L., GHEORGHES, T. N., AND GALATSIDAS, M. Do observers like curvature or do they dislike angularity? *British Journal of Psychology 107*, 1 (2016), 154–178. doi:10.1111/bjop.12132.

[4] BEVEN, K. J., AND KIRKBY, M. J. A physically based, variable contributing area model of basin hydrology/un modèle à base physique de zone d'appel variable de l'hydrologie du bassin versant. *Hydrological Sciences Journal 24*, 1 (1979), 43–69. doi:10.1080/02626667909491834.

[5] BRIGGER, P., HOEG, J., AND UNSER, M. B-spline snakes: a flexible tool for parametric contour detection. *IEEE Transactions on image processing 9*, 9 (2000), 1484–1496. doi:10.1109/83.862624.

[6] BRUNET, P., AND PEREZ-VIDAL, L. Smooth contour line construction with spline interpolation. *Questiió: Quaderns d'Estadística, Sistemes, Informatica i Investigació Operativa 8*, 3 (1984).

[7] BUTTENFIELD, B. Treatment of the cartographic line. *Cartographica: The International Journal for Geographic Information and Geovisualization 22*, 2 (1985), 1–26. doi:10.3138/FWV8-3802-2282-6U47.

[8] CHAIKIN, G. M. An algorithm for high-speed curve generation. *Computer graphics and image processing 3*, 4 (1974), 346–349. doi:10.1016/0146-664X(74)90028-8.

[9] CHRISTENSEN, A. H. Contour smoothing by an eclectic procedure. *Photogrammetric engineering and remote sensing 67*, 4 (2001), 511–517.

[10] DIBIASE, D. Contouring by hand. In *Nature of Geographic Information*. The Pennsylvania State University, 2014. https://www.e-education.psu.edu/natureofgeoinfo/c7_p6.html.

[11] DIETRICH, W. E., BELLUGI, D. G., SKLAR, L. S., STOCK, J. D., HEIMSATH, A. M., AND ROERING, J. J. Geomorphic transport laws for predicting landscape form and dynamics. *Geophysical Monograph-American Geophysical Union 135* (2003), 103–132. doi:10.1029/135GM09.

[12] DOUGLAS, D. H., AND PEUCKER, T. K. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: The International Journal for Geographic Information and Geovisualization 10*, 2 (1973), 112–122. doi:10.1002/9780470669488.ch2.

[13] DUTTON, G. H. Fractal enhancement of cartographic line detail. *The American Cartographer 8*, 1 (1981), 23–40. doi:10.1559/152304081783948430.

[14] EPA. Guidance for geospatial data quality assurance project plans EPA: QA/G-5G. Tech. rep., US Environmental Protection Agency Office of Environmental Information, 2003.

[15] GITLOW, H., MELNYCK, R., AND LEVINE, D. A guide to six sigma and process improvement for practitioners and students: Foundations. *DMAIC, Tools, Cases, and Certification, Second Edition, 2nd edition, PH Professional Business* (2015).

[16] GÖKGÖZ, T. Generalization of contours using deviation angles and error bands. *The Cartographic Journal 42*, 2 (2005), 145–156. doi:10.1179/000870405X61441.

[17] GÖKGÖZ, T., AND GÜLGEN, F. Comparison of two methods for deriving skeleton lines of terrain. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 34* (1991).

[18] GÖKGÖZ, T., AND SELÇUK, M. A new approach for the simplification of contours. *Cartographica: The International Journal for Geographic Information and Geovisualization 39*, 4 (2004), 37–44.

[19] GOODCHILD, M. F., AND HUNTER, G. J. A simple positional accuracy measure for linear features. *International Journal of Geographical Information Science 11*, 3 (1997), 299–306. doi:10.1080/136588197242419.

[20] GOULDEN, T. Prediction of error due to terrain slope in LiDAR observations. Tech. Rep. 265, Geodesy and Geomatics Engineering, University of New Brunswick, 2009.

[21] GUILBERT, E., GAFFURI, J., AND JENNY, B. Terrain generalisation. In *Abstracting geographic information in a data rich world: methodologies and applications of map generalisation*. Springer, 2014, pp. 227–258. doi:10.1007/978-3-319-00203-3_8.

[22] GUILBERT, E., AND SAUX, E. Cartographic generalisation of lines based on a b-spline snake model. *International Journal of Geographical Information Science 22*, 8 (2008), 847–870.

[23] HODGSON, M. E., AND BRESNAHAN, P. Accuracy of airborne lidar-derived elevation. *Photogrammetric Engineering & Remote Sensing 70*, 3 (2004), 331–339. doi:10.14358/PERS.70.3.331.

[24] IMHOF, E. *Cartographic Relief Presentation*. Walter de Gruyter, 1982.

[25] ISENBERG, M. LASTools Documentation: LiDAR to DEM: lasgrid_README.txt, 2020. https://downloads.rapidlasso.de/lasgrid_README.txt.

[26] JAMES, L. A., WATSON, D. G., AND HANSEN, W. F. Using LiDAR data to map gullies and headwater streams under forest canopy: South Carolina, USA. *Catena 71*, 1 (2007), 132–144. doi:10.1016/j.catena.2006.10.010.

[27] JENSON, S. K. Applications of hydrologic information automatically extracted from digital elevation models. *Hydrological Processes 5*, 1 (1991), 31–44. doi:10.1002/hyp.3360050104.k.

[28] JENSON, S. K., AND DOMINGUE, J. O. Extracting topographic structure from digital elevation data for geographic information system analysis. *Photogrammetric engineering and remote sensing 54*, 11 (1988), 1593–1600.

[29] KETTUNEN, P., KOSKI, C., AND OKSANEN, J. A design of contour generation for topographic maps with adaptive DEM smoothing. *International Journal of Cartography 3*, 1 (2017), 19–30. doi: 10.1080/23729333.2017.1300998.

[30] KLIPPEL, A., BAGHER, M., AND WALGRUN, J. PSU Online Course: Geography 497 Modeling and Virtual Reality, Section 3.2 Smoothing the DEM and Creating Contours, 2022. https://www.e-education.psu.edu/geogvr/node/848.

[31] LAZZARI, M. High-resolution LiDAR-derived DEMs in hydrografic network extraction and short-time landscape changes. In *Computational Science and Its Applications–ICCSA 2020: 20th International Conference, Cagliari, Italy, July 1–4, 2020, Proceedings, Part II 20* (2020), Springer, pp. 723–737. doi:10.1007/978-3-030-58802-1_52.

[32] LI, Z., AND SUI, H. An integrated technique for automated generalization of contour maps. *The Cartographic Journal 37*, 1 (2000), 29–37. doi:10.1179/caj.2000.37.1.29.

[33] LINDSAY, J. B., FRANCIONI, A., AND COCKBURN, J. M. LiDAR DEM smoothing and the preservation of drainage features. *Remote Sensing 11*, 16 (2019), 1926. doi:10.3390/rs11161926.

[34] LINDSAY, J. B., NEWMAN, D. R., AND FRANCIONI, A. Scale-optimized surface roughness for topographic analysis. *Geosciences 9*, 7 (2019), 322. doi:10.3390/geosciences9070322.

[35] MAUNE, D. F. *Digital Elevation Model Technologies and Applications: The DEM user Manual*, 2nd ed. ASPRS Publications, 2007.

[36] MCCULLAGH, M. J. Creation of smooth contours over irregularly distributed data using local surface patches. *Geographical Analysis 13*, 1 (1981), 51–63. doi:10.1111/j.1538-4632.1981.tb00714.x.

[37] MCMASTER, R. B. The geometric properties of numerical generalization. *Geographical Analysis 19*, 4 (1987), 330–346. doi:10.1111/j.1538-4632.1987.tb00134.x.

[38] MCMASTER, R. B., AND SHEA, K. S. Generalization in digital cartography. Association of American Geographers.

[39] MEKONNEN, B. A., NAZEMI, A., MAZUREK, K. A., ELSHORBAGY, A., AND PUTZ, G. Hybrid modelling approach to prairie hydrology: fusing data-driven and process-based hydrological models. *Hydrological Sciences Journal 60*, 9 (2015), 1473–1489. doi:10.1080/02626667.2014.935778.

[40] MEKONNEN, B. A., NAZEMI, A., MAZUREK, K. A., ELSHORBAGY, A., AND PUTZ, G. Hybrid modelling approach to prairie hydrology: fusing data-driven and process-based hydrological models. *Hydrological Sciences Journal 60*, 9 (2015), 1473–1489. doi:10.1080/02626667.2014.935778.

[41] MINÁR, J., AND EVANS, I. S. Elementary forms for land surface segmentation: The theoretical basis of terrain analysis and geomorphological mapping. *Geomorphology 95*, 3-4 (2008), 236–259. doi:10.1016/j.geomorph.2007.06.003.

[42] MOZAS, A. T., AND ARIZA, F. J. Methodology for positional quality control in cartography using linear features. *The Cartographic Journal 47*, 4 (2010), 371–378. doi:10.1179/000870410X12825500202931.

[43] MOZAS-CALVACHE, A. T., UREÑA-CÁMARA, M. A., AND PÉREZ-GARCÍA, J. L. Accuracy of contour lines using 3d bands. *International Journal of Geographical Information Science 27*, 12 (2013), 2362–2374. doi:10.1080/13658816.2013.801484.

[44] NATIONAL GEOSPATIAL INTELLIGENCE AGENCY (NGA). MIL-T-89301A, 1995. http://everyspec.com/MIL-SPECS/MIL-SPECS-MIL-T/MIL-T-89301A_26013/.

[45] PASDA (PENNSYLVANIA SPATIAL DATA ACCESS). Geospatial portal, 2017. https://www.pasda.psu.edu/uci/FullMetadataDisplay.aspx?file=USGS_LiDAR2017.xml.

[46] PIKE, R. J., EVANS, I., AND HENGL, T. Geomorphometry: a brief guide. *Developments in soil science 33* (2009), 3–30. doi:10.1016/S0166-2481(08)00001-9.

[47] QUINN, P., BEVEN, K., CHEVALLIER, P., AND PLANCHON, O. The prediction of hillslope flow paths for distributed hydrological modelling using digital terrain models. *Hydrological Processes 5*, 1 (1991), 59–79.

[48] SHANNON, C. E. A mathematical theory of communication. *The Bell system technical journal 27*, 3 (1948), 379–423.

[49] SOFIA, G. Combining geomorphometry, feature extraction techniques and earth-surface processes research: The way forward. *Geomorphology 355* (2020), 107055. doi:10.1016/j.geomorph.2020.107055.

[50] STOKER, J., AND MILLER, B. The accuracy and consistency of 3d elevation program data: A systematic analysis. *Remote Sensing 14*, 4 (2022), 940. doi:10.3390/rs14040940.

[51] THOMAS, I., JORDAN, P., SHINE, O., FENTON, O., MELLANDER, P.-E., DUNLOP, P., AND MURPHY, P. N. Defining optimal DEM resolutions and point densities for modelling hydrologically sensitive areas in agricultural catchments dominated by microtopography. *International Journal of Applied Earth Observation and Geoinformation 54* (2017), 38–52. doi:10.1016/j.jag.2016.08.012.

[52] TYLER, D. J., AND GREENLEE, S. K. Creation of digital contours that approach the characteristics of cartographic contours: U.S. Geological Survey Scientific Investigations Report 2012–5167. Tech. rep., US Geological Survey, 2012. https://pubs.usgs.gov/sir/2012/5167/sir2012-5167.pdf.

[53] USGS 3DEP. Lidar base specification online, 2024. https://www.usgs.gov/ngp-standards-and-specifications/lidar-base-specification-online.

[54] VEREGIN, H. A taxonomy of error in spatial databases. technical paper 89–12. Tech. rep., National Center for Geographic Information and Analysis, University of California, Santa Barbara, California, USA, 1989. http://ncgia.ucsb.edu/technical-reports/PDF/89-12.pdf.

[55] WALPOLE, R. E., MYERS, R. H., MYERS, S. L., AND YE, K. *Probability and statistics for engineers and scientists*, vol. 6. Prentice Hall, 1998.

[56] WARE, J., AND JONES, C. B. A spatial model for detecting (and resolving) conflict caused by scale reduction. *Advances in GIS Research 2* (1996), 547–558.

[57] WARE, J. M. A procedure for automatically correcting invalid flat triangles occurring in triangulated contour data. *Computers & Geosciences 24*, 2 (1998), 141–150. doi:10.1016/S0098-3004(97)00088-5.

[58] WARE, J. M., JONES, C. B., AND BUNDY, G. L. A triangulated spatial model for cartographic generalisation of areal objects. In *Spatial Information Theory, A Theoretical Basis for GIS: International Conference COSIT'95 Semmering, Austria, September 21–23, 1995 Proceedings* (1995), Springer, pp. 173–192. doi:10.1007/3-540-60392-1_12.

[59] WATSON, D. *Contouring: a guide to the analysis and display of spatial data*. Elsevier, 2013. doi:10.1016/0098-3004(93)90069-H.

[60] WEIBEL, R. Models and experiments for adaptive computer-assisted terrain generalization. *Cartography and Geographic Information Systems 19*, 3 (1992), 133–153. doi:10.1559/152304092783762317.

[61] WEIBEL, R., AND BURGARDT, D. On-the-fly generalization. In *Encyclopedia of GIS*, S. Shashi and X. Hui, Eds. Springer, 2008, pp. 339–344. doi:10.1007/978-0-387-35973-1_450.

[62] WEIBEL, R., AND DUTTON, G. Generalising spatial data and dealing with multiple representations. *Geographical information systems 1* (1999), 125–155.

[63] WILSON, J. P. Geomorphometry: Today and tomorrow. *PeerJ Preprints 6* (2018), e27197v1. doi:10.7287/peerj.preprints.27197v1.

[64] ZHANG, J., WU, J., CHANG, K., ELLIOT, W., AND DUN, S. Effects of DEM source and resolution on WEPP hydrologic and erosion simulation: a case study of two forest watersheds in northern Idaho. *Transactions of the ASABE 52*, 2 (2009), 447–457.

A A review of devised testing methods

Data Set Label	Description
DEM	The unaltered LiDAR DEM was used to generate *NICL Baseline* and, for tests 11-14 in Table A-2, estimates of true height at vertices along a smoothed contour line are compared to the nominal height of the contour line.
NICL Baseline	The original NICL contours were generated from the unaltered DEM and applied without thinning or smoothing. Small closed polylines have been culled.
NICL Thin 1.2m	A contour data set generated from *NICL Baseline* thinned at tolerance T_T = 1.2 meters. This is input to the process that generates *LACA 6K Smooth* and *B-Spline Snakes* contours.
NICL Buffer 0.6m	The 0.6-meter buffer about *NICL Baseline* centerlines. Used to determine inside or out for contour x, y accuracy.
NICL Buffer 1.2m	The 1.2-meter buffer about *NICL Baseline* centerlines. Used to determine inside or out for contour x, y accuracy.
NICL Thin 0.5m	A special selection of the *NICL Baseline* data thinned at T_T = 0.5 m for Test 17. Used to minimally thin the data to reveal the angularity which was masked by the oversampled points in the the *NICL Baseline*.
5X Smoothing	Contours are generated from an intermediate smoothed DEM by simple averaging with five iterations of smoothing for the original DEM.
B-Spline Snakes	A contour data set generated by processing *NICL Thin 1.2m* through a *B-Spline Snakes* line generalization function available through QGIS 3.16. The test used default shape parameters (Alpha = 1.0 and Beta = 0.5).
Kettunen FPDEMS	An intelligent, "feature preserving DEM smoothing" (FPDEMS) method alternately using "heavy" (we used 5X iterations) versus "light" (1X iteration) smoothed values weighted based on the topographic position index value at the center point. *Kettunen FPDEMS* contours are then generated using a composite intermediate DEM that, in turn, is derived from two other intermediate DEMs, one based on heavy smoothing (5X) and the other on light smoothing (1X).
LACA 6K Smooth	The smoothed LACA contours at a 1:6,000 scale were generated by smoothing *NICL Thin 1.2m* using a 0.6-meter value for I_T. This is our preferred method of contour smoothing.
LACA 50K Smooth	A smoothed LACA contour data set at 1:50,000 scale was generated from NICL Baseline using T_T = 50,000 * 0.2 /1000 = 10 meters and I_T = 5 meters to generate new smoothed points (see Equation 1 in Section 2.4.2.1).This data is intended to demonstrate the applicability of the LACA method at medium scales.

Table A-1: Data set labels and descriptions.

Test		No.	Test Method	Data Set Evaluated	Reference Data Set
Horizontal Accuracy	Method 1	1	Buffer Pct In	*LACA 6K Smooth*	*NICL Buffer 0.6m*
		2	Buffer Pct In	*LACA 6K Smooth*	*NICL Buffer 1.2m*
		3	Buffer Pct In	*B-Spline Snakes*	*NICL Buffer 0.6m*
		4	Buffer Pct In	*B-Spline Snakes*	*NICL Buffer 1.2m*
		5	Buffer Pct In	*5X Smoothing*	*NICL Buffer 0.6m*
		6	Buffer Pct In	*5X Smoothing*	*NICL Buffer 1.2m*
		7	Buffer Pct In	*Kettunen FPDEMS*	*NICL Buffer 0.6m*
	Method 2	8	Buffer Pct In	*Kettunen FPDEMS NICL*	*NICL Buffer 1.2m*
		9	Buffer Pct In	*LACA 50K Smooth*	*NICL Buffer 0.6m*
		10	Buffer Pct In	*LACA 50K Smooth*	*NICL Buffer 1.2m*
		11	Hypothesis Testing I	*LACA 6K Smooth*	*NICL Baseline*
		12	Hypothesis Testing I	*B-Spline Snakes*	*NICL Baseline*
		13	Hypothesis Testing I	*5X Smoothing*	*NICL Baseline*
		14	Hypothesis Testing I	*Kettunen FPDEMS*	*NICL Baseline*
		15	Hypothesis Testing I	*LACA 50K Smooth*	*NICL Baseline*
Smoothness		16	Hypothesis Testing II	*LACA 6K Smooth*	*NICL Baseline*
		17	Hypothesis Testing II	*LACA 6K Smooth*	*NICL Thin 0.5m*
		18	Hypothesis Testing II	*B-Spline Snakes*	*NICL Thin 0.5m*
		19	Hypothesis Testing II	*B-Spline Snakes*	*LACA 6K Smooth*
		20	Hypothesis Testing II	*5X Smoothing*	*NICL Thin 0.5m*
		21	Hypothesis Testing II	*5X Smoothing*	*LACA 6K Smooth*
		22	Hypothesis Testing II	*Kettunen FPDEMS*	*NICL Thin 0.5m*
		23	Hypothesis Testing II	*Kettunen FPDEMS*	*LACA 6K Smooth*
Symmetry		24	By Angle	*LACA 6K Smooth*	Identity Line
		25	By Projected Distance	*LACA 6K Smooth*	Identity Line

Table A-2: Inputs to tests conducted in this study.

B A data structure and algorithm for removing self-coalesced line sections

In light of the disappointing results for 1:50,000 scale smoothing, we must acknowledge the need for further methods of generalization when affecting a significant scale change. To explore such a solution, we speculate here on a path for automated generalization. We describe a data structure capable of supporting the automated generalization of contours and an algorithm that mimics manual simplification of the lines by automatically eliminating coalesced sections of the line. Refer to the basic structures in Figure B-1 and the rudimentary algorithm described in pseudocode in the Appendix C and here in Figures B-2 to B-4.

B.1 A topological TIN data structure for generalization

Most triangle structures for TINs are centered on a three-point sequence as the basic element. These are simple structures usually designed for minimum storage and easy display. Other elements of a triangulation, especially edges between two triangles (hence the term *topological*), are seldom explicitly manifest as elements in the TIN. They can only be deduced through search methods. In a series of papers, Ware et al. describe a "Simplicial

1	Method 1: Testing of X and Y Accuracy by Buffer Percent Inside	A test of x and y accuracy based on the count of points inside the buffer (either 0.6 or 1.2 m) divided by the total count for the data set being tested. Criteria for accepting the test results are based on established percentages of points adapted from quality assurance testing. See Section 2.5.
2	Method 2: Hypothesis Testing I: Tests of X and Y Accuracy Inferred from Height Differences	The first version of the hypothesis test used here assesses the x and y accuracy by inference of the difference in height between accurate estimates from the DEM and the nominal height of the contour line. See Section 2.5 and Figure 3. This is a test of two means using a null hypothesis that the means are effectively equal; that there is no difference after treatment of the input data set. Here, acceptance of the null means that the smoothed line is accurate and adheres closely to the *NICL Baseline* contours. Rejection of the null hypothesis indicates that the smoothed line departs significantly in x and y from *NICL Baseline* contours; that it is inaccurate.
3	Hypothesis Testing II: Tests of Smoothness of the Lines	The second version of the hypothesis test used here assesses the amount of angularity removed by smoothing. This test is based on the notion of an enclosed angle associated with each interior vertex in a line. See Section 2.5.4. Acceptance of the null hypothesis here means that the process being evaluated has not significantly changed the overall smoothness compared to the reference line. A rejection of the null means that the process that created the data set being evaluated has reduced the angularity and made the line smoother.
4	Regression Analysis and Significance Testing for Symmetry in the *LACA 6K Smooth* Contours	Linear regression is an ideal method for testing symmetry of the *LACA 6K Smooth* results. A resulting equation of y = x indicates perfect symmetry. By using the medial axis along the angle bisector, an interval can be divided into two halves and measures of smoothed points in one half (i.e., the *previous* half) can be compared to corresponding points in the next half. See Section 2.5.5 and Figures 6 and 7. A t-test is the preferred statistic for our test of significance. Acceptance of a null hypothesis indicates no linear relation can be found, while rejection of the null indicates that a linear relation exists. The closer the slope of the regression line is precisely to 1 is an indication of relative success of the method.

Table A-3: Test method, description, and criteria for success and failure.

Data Structure" that more fully enumerates relations among potential TIN model elements, including explicitly defined edges [56–58].

While an edge at first glance seems to be redundant, it is, in fact, an efficient element for search and analysis and for using a triangulated model for anything other than efficient storage and ease of display [58]. For example, basic navigation through the TIN is facilitated by recursively exploiting *edge-to-triangle-to-edge* relations as we describe in the

A Proposed Topological TIN Model Structure

```
struct CtrVtx                                   // A Contour Line Vertex
{
    double x;                                   // Geometry for the Contour Vertex
    double y;
    double hgt;
    struct CtrLine *OwnerLine;                  // The Contour Polyline containing this CtrVtx
    struct Edge *OwnerEdge;                     // An Edge containing this CtrVtx in the TIN
    struct CtrVtx *NextVtx;                     // Linked list of pointers to neighboring Contour Vertices along the Contour Line
    struct CtrVtx *PrevVtx;
    long Prop;                                  // Property Bit-Mask
};

struct Tri                                      // A Triangle structure with explicit Edge pointers
{
    struct Edge *Edge1;                         // Flagged Edges 1-3 of the Triangle
    struct Edge *Edge2;
    struct Edge *Edge3;
    long FlipBits;                              // Flags for flipped Edges 1-3 with respect to enforced
};                                              // CCW orientation of each Triangle

struct Edge                                     // A TIN Edge with From and To Vertex and Left and Right Triangle Pointers
{
    struct CtrVtx *FromVtx;                     // From and To Vertices for directed Edges
    struct CtrVtx *ToVtx;
    struct Triangle *TriLeft;                   // Left & Right Triangles (with respect to Edge->From to  Edge->To direction)
    struct Triangle *TriRight;
};

struct CtrLine                                  // A Contour Polyline
{
    double CtrHgt;                              // Height of the Contour Polyline
    struct CtrVtx *RootVtx;                     // Root and Terminal Vertices for the CtrLine
    struct CtrVtx *TermVtx;
};
```

Figure B-1: A topological TIN data structure for contour line generalization.

structures above and the algorithm outlined in Appendix C (see the combination at work in the REMOVE_SELF_COALESCED_LINE_SECTIONS and GET_NEXT_PIVOT_EDGE functions in the pseudocode).

With regard to the structures, we also propose adding a vertex-to-polyline link (i.e., OwnerLine) and an explicit link of the triangle to constituent edges (i.e., Edges 1 to 3). This includes bitmask flags associated with the individual edge1- to -3-pointers to indicate whether or not to reverse the natural "from-" and "to-" orientation of the edge. We do this in order to be able to form a consistent CCW-oriented triangle; a seemingly arbitrary but practically useful convention.

Finally, we should note that the structure includes a convention of directed ordering of the vertices in triangle edges. Successive vertices from a contour line are used to form a directed feature edge using the from and to elements of the edge structure. Also, edges formed between two vertices from the same contour line are ordered such that the vertex occurring first in sequence is the from vertex in the edge (see Figure B-4A). These conventions fortify the structures and simplify the development of the algorithm.

B.2 An algorithm for resolving self-coalesced line sections

Here we present an algorithm for removing vertices involved in a coalesced section of the line using the structures defined in Figure B-1. The discussion is keyed to Figures B-2 to B-5.

In Figure B-2, the triangulated contours for *LACA 50K Smooth* are shown. The coalesced sections of the lines are highlighted in the figure and annotated as *coalesced areas* (CA_{k-m}). Each CA is identified by searching for edges along the contour lines that are: 1) common

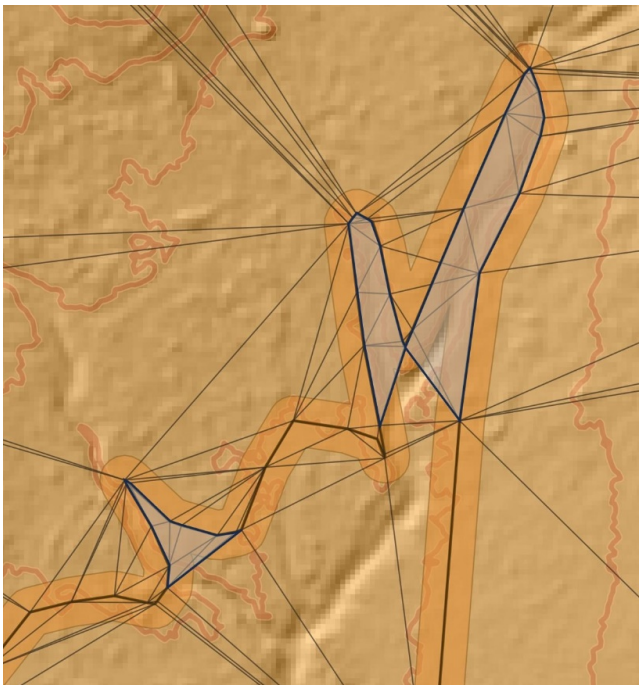


Figure B-2: Coalesced areas and originating edges.

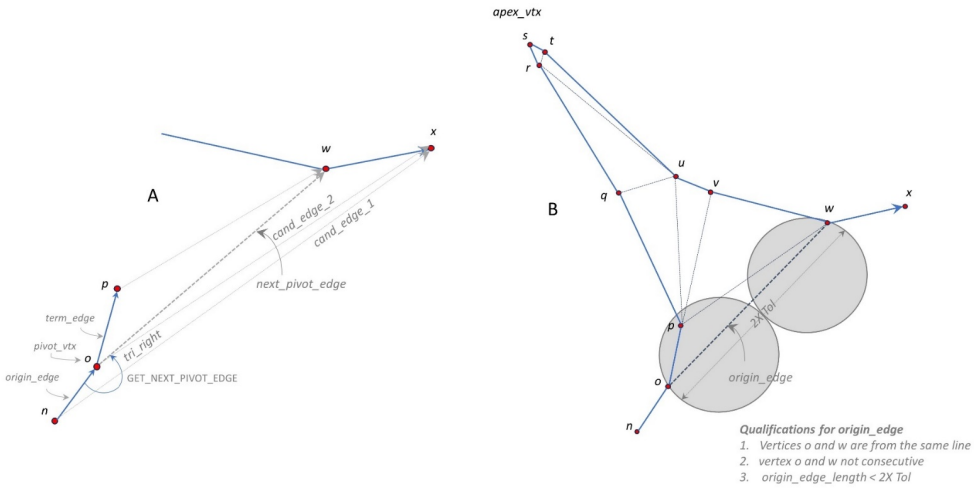


Figure B-3: Navigating a topological TIN network (A) and qualifications for an originating Edge (B).

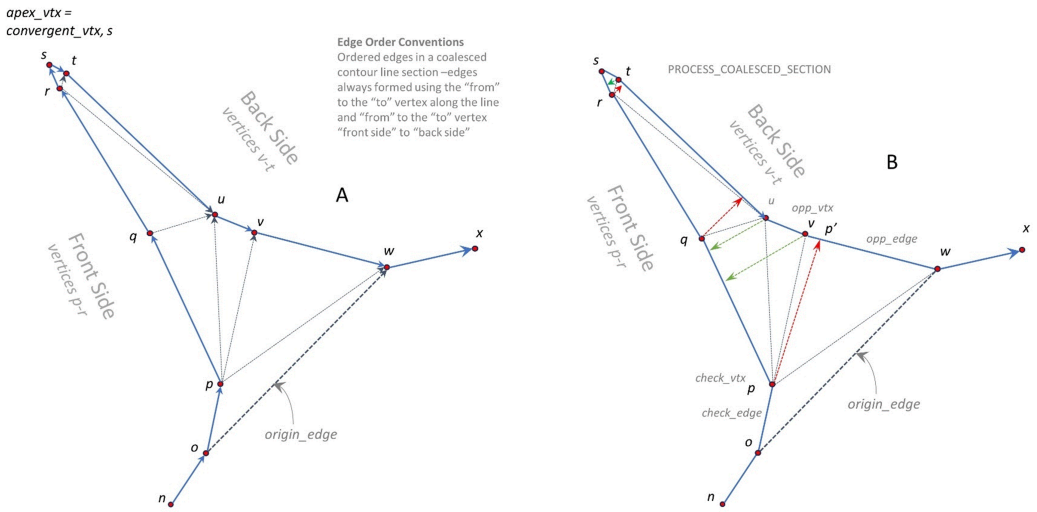


Figure B-4: Edge order conventions (A) and projecting vertices perpendicularly onto opposing edges (B)

to a single owner line; 2) composed of two non-consecutive vertices in that line; and, 3) less than two times the line thickness in length or projected perpendicular distance. These are *originating edges* (OE_{k-m}), that is, an edge that begins the definition of an enclosed, self-coalesced section of a contour polyline.

The vertices and edges are elements involved in traversing the TIN and are described in the enlargement of area CA_k in Figure B-3. In Figure B-3A, from a current pivot vertex (*pivot_vtx*) and *origin_edge*, till the terminating edge (*term_edge*) is encountered, a pivot operation about *pivot_vtx* is executed. The triangle right of the edge is identified as a candidate edge and a prospective *origin_edge*. We can identify the *origin_edge* as one of three candidates that make up *tri_right*. We know that the *pivot_vtx* is common to two candidate edges, only one of which is the *next_pivot_edge*, determined by matching endpoints of all involved edges. The line is exhausted by a search of the right side of the line in a series of CCW pivots up to the *apex_vertex* identified as vertex s in Figure B-3B. The *next_pivot_edge* is also identified in Figure B-4A. A check of the length of that edge in Figure B-3B identifies whether or not *next_pivot_edge* is the *origin_edge* for CA_k.

In Figure B-4A, the *ordering conventions* for the contours enmeshed in a TIN are shown. The vertex-to-owner-edge relations for vertices in a line all point to an edge connecting the present vertex to the next one in sequence in the polyline. Additionally, vertices in the coalesced area are directed from the *front side* of the coalesced section (vertices o through r) to the *back side* (vertices w through t). In Figure B-4B, the projected points for successive vertices are shown. These projected points, such as p' in Figure B-4B, define the closest distance between the points and opposing edges.

In Figure B-4B, the process of finding the shortest distance from a vertex being tested for coalescence (*check_vtx*) to an opposing edge (*opp_edge*) is illustrated. Normally, the perpendicular distance is the shortest and is characrized by $0.0 < rp < 1.0$. However, under conditions where the projected value for successive edge checks abruptly change from

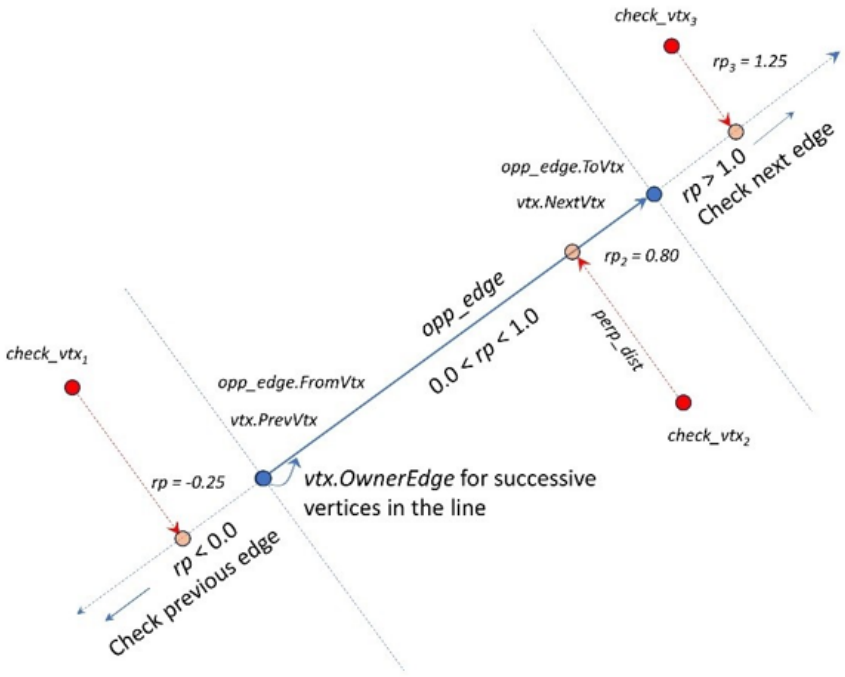


Figure B-5: Relative position (rp) of projected points along an opposing edge. The *check_vtx1* projects off the line segment at an $rp = -0.25$: check the previous edge. The *check_vtx2* projects at 0.80: the shortest distance between the *check_vtx* and *opp_edge* is the perpendicular distance. The *check_vtx3* projects at an $rp = 1.25$, beyond the furtherest extent of the segment: check the next edge.

$rp > 1.0$ to $rp < 0.0$, the directly connected edge between *check_vtx* and a vertex of the *opp_edge* is the minimum distance.

The *relative position* (rp) of the projected point (p' in Figure B-4B) is used as a test to determine whether or not the perpendicular projects within the extent of the opposite edge ($0 < rp < 1.0$). Figure B-5 describes possible relations between a *check_vtx* and an *opp_edge* and how these relations drive further checking. The concept of a projected point on a line segment is used in several methods and tests in this paper. Here, if the the perpendicular distance between the *pivot_vtx* and *opp_edge* is found to be less than two times the line thickness tolerance, the vertex can be marked for deletion.

After a line is evaluated, a second pass through the vertices deletes marked vertices from the doubly-linked list used to represent the line. In this way, the line is more properly generalized.

Refer to the pseudocode for the algorithm in Appendix C for further details.

C Algorithm for removing self-coalesced sections in contour lines

```
function REMOVE_SELF_COALESCED_LINE_SECTIONS(root_ctr_vtx, dir)
    root_ctr_vtx    //   Root pointer to a doubly-linked list of vertices that make up the line
    dir             //   Direction of search, RIGHT or LEFT side of the line - this function must be called once for each direction
    edge = root_ctr_vtx.OwnerEdge    //  For contor lines, an owner edge is always created between vtx i to vtx i+1
    term_edge = root_ctr_vtx.NextVtx.OwnerEdge   // Works since edge vertices in the line are ordered vtx i to vtx i+1 and
    pivot_vtx = term_edge.FromVtx                // at least three vertices in the line
    if dir = RIGHT
        tri = edge.TriRight        // Begin with known triangle right of line
    else
        tri = edge.TriLeft         // Subsequent operations remain on the right or left side of the line
    endif
    do                             // Do-While there are still unchecked vertices in the line
        while edge <> term_edge    // Pivot CCW for Front Side or CW for Back Side about the edge.ToVtx
            if edge = tri.Edge1    // This edge is the first edge in the current triangle, check 2 and 3 for next pivot
                edge.GET_NEXT_PIVOT_EDGE (tri. Edge2, tri.Edge3, pivot_vtx, vtx_opp_edge, next_pivot_edge)
                new_tri = next_pivot_edge.LeftTri
                if next_pivot_edge.LeftTri = tri
                    new_tri = next_pivot_edge.RightTri    // Deduce the new triangle
                endif
            else if edge = tri.Edge2           // This edge is the second edge in the current triangle, check1 and 3 for next pivot
                edge.GET_NEXT_PIVOT_EDGE (tri.Edge1, tri.Edge3, pivot_vtx, vtx_opp_edge, next_pivot_edge)
                new_tri = next_pivot_edge.LeftTri
                if next_pivot_edge.LeftTri = tri
                    new_tri = next_pivot_edge.RightTri
                endif
            else                           // This edge must be the third edge in the current triangle, check 1 and 2 for next pivot
                edge.GET_NEXT_PIVOT_EDGE (tri.Edge1, tri.Edge2, pivot_vtx, vtx_opp_edge, next_pivot_edge)
                new_tri = next_pivot_edge.LeftTri
                if new_tri = tri                       // Deduce the new triangle - left or right
                    new_tri = next_pivot_edge.RightTri
                end if
            end if
            if pivot_vtx.OwnerLine = vtx_opp_edge.OwnerLine              // Vertices from the same line?
                if edge.NextVtx <> vtx_opp_edge                          // And the vertices are not consecutive?
                    if PERP_DIST(edge.FromVtx, vtx_opp_edge) < 2X_Tol    // Tol less than 2X contour line thickness?
                        origin_edge = edge                               // Found beginning of coalesced section
                    endif
                endif
            endif
            edge = next_pivot_edge
            tri = new_tri
        end while
        pivot_vtx = pivot_vtx.NextVtx            // Works because all pivot vertices are consecutive along the line
        term_edge = term_edge.ToVtx.OwnerEdge    // Works because the edge in the line is ordered vtx i to vtx i+1
        if dir = RIGHT                           // Reset the triangle by checking RIGHT or LEFT search direction
            tri = term_edge.TriRight
        else
            tri = term_edge.TriLeft
        endif
    while pivot_vtx <> NULL                    // List of coordinates is exhausted when pivot_vtx is NULL
    UPDATE_CONTOUR_LINE(root_ctr_vtx)          // Unlink the vertices marked to delete the coalesced sections
end function
```

```
function edge.GET_NEXT_PIVOT_EDGE(first_cand_edge, second_cand_edge, pivot_vtx, vtx_opp_edge, next_pivot_edge)
    // Compares the alternative two edges about a pivot vertex in a triangle to determine
    // the opposite vertex and next CW or CCW pivot edge
    if pivot_vtx = first_cand_edge.FromVtx    // All checks are against the term edge from vertex, the pivot vertex
          vtx_opp_edge = first_cand_edge.ToVtx
          next_pivot_edge = first_cand_edge
       else if pivot_vtx= second_cand_edge.FromVtx
          vtx_opp_edge = second_cand_edge.ToVtx
          next_pivot_edge = second_cand_edge
       endif
    end function

    function PROCESS_COALESCED_SECTION(dir, origin_edge)
       check_edge = origin_edge.FromVtx.OwnerEdge     // Convoluted, but precise
       check_vtx = check_ edge.ToVtx                  // First vertex along the line past originating edge from vertex
       convergent_vtx = NULL                          // The convergent vertex is the apex of a simple coalesced section
                                                      // Apex:  a crenulation vertex where consecutive edges point to the same triangle
       while convergent_vtx = NULL                    // First, find the convergent (apex) vertex
          if dir = RIGHT
             if check_vtx.OwnerEdge.TriRight = check_vtx.NextVtx.OwnerEdge.TriRight  // Consecutive edges point to the same triangle?
                convergent_vtx = check_vtx.NextVtx
             end if
          else                                        // Searching LEFT of line
             if check_vtx.OwnerEdge.TriLeft = check_vtx.NextVtx.OwnerEdge.TriLeft    // Consecutive edges point to the same triangle?
                convergent_vtx = check_vtx.NextVtx
             end if
          end if
             if convergent_edge = NULL
             check_vtx = check_vtx.NextVtx           // Keep searching
          end if
       end while
       check_edge = origin_edge.FromVtx.OwnerEdge    // Process the front side of the coalesced section,
       check_vtx = check_ edge.ToVtx
       opp_edge = origin_edge.ToVtx.PrevVtx.OwnerEdge
       opp_vtx = opp_ edge.ToVtx
       while check_vtx <> convergent_vtx             // Vertices in the coalesced section have not yet converged at the apex
          PROJ_VTX_TO_EDGE(check_vtx, opp_edge, perp_dist, x_proj, y_proj, rp)
          if rp > 0.0 & rp < 1                       // Projects within the extent of the opposite edge?
             if perp_dist  < 2X_Tol                  // Compare check vertex to opposite edge using 2X tolerance
                check_vtx.Prop = MARK_TO_DELETE      // Mark for deletion
             end if
          else if rp < 0.0                           // Check vertex projects  "below" the from vertex of the edge
             while rp < 0.0                          // Bypass edges till we find one that projects > 0.0
                opp_edge = opp_vtx.FromVtx.OwnerEdge
                opp_vtx = opp_vtx.FromVtx
                PROJ_VTX_TO_EDGE(check_vtx, opp_edge, perp_dist, x_proj, y_proj, rp)
                if rp > 1.0    // Traps the case where rp switches abruptly from "below" to "above" for the to vertex of the edge
                               // This alignment means that the check to opposite vertices is the shortest distance
                   perp_dist = SQRT((opp_vtx.x - check_edge.x)2 + (opp_vtx.y - check_edge.y)2)
                   if perp_dist  < 2X_Tol              // Compare check vertex to opposite edge using 2X tolerance
                      check_vtx.Prop = MARK_TO_DELETE  // Mark for deletion
                   end if
                end if
             end while
          end if
          check_vtx = check_vtx.NextVtx
       end while
```

```
    check_edge = origin_edge.ToVtx.PrevVtx.OwnerEdge     // Process the backside vertices in the coalesced section
    check_vtx = check_ edge_left.ToVtx
    opp_edge = origin_edge.FromVtx.OwnerEdge
    opp_vtx = opp_ edge.FromVtx
    while check_vtx <> convergent_vtx                    // Vertices in the coalesced section have not yet converged at the apex
        PROJ_VTX_TO_EDGE(check_vtx, opp_edge, perp_dist, x_proj, y_proj, rp)
        if rp > 0.0 & rp < 1                             // Projects within the extent of the opposite edge?
            if perp_dist < 2X_Tol                        // Compare check vertex to opposite edge using 2X tolerance
                check_vtx.Prop = MARK_TO_DELETE          // Mark for deletion
            end if
        else if rp > 1.0                                 // Check vertex projects "below" the from vertex of the edge
            while rp > 1.0                               // Bypass edges till we find one that projects > 0.0
                opp_edge = opp_vtx.NextVtx.OwnerEdge
                opp_vtx = opp_edge.FromVtx
                PROJ_VTX_TO_EDGE(check_vtx, opp_edge, perp_dist, x_proj, y_proj, rp)
                if rp < 0.0      // Traps the case where rp switches abruptly from "below" to "above" for the to vertex of the edge
                                 // This alignment means that the edge from check to opposite vertices is shortest distance - neither
                                 // perpendicular test for successive edges falls within the 0 to 1 range
                    perp_dist = SQRT((opp_vtx.x - check_edge.x)2 + (opp_vtx.y - check_edge.y)2)
                    if perp_dist < 2X_Tol                // Compare check vertex to opposite edge using 2X tolerance
                        check_vtx.Prop = MARK_TO_DELETE  // Mark for deletion
                    end if
                end if
            end while
        end if
        check_vtx = check_vtx.NextVtx
    end while
    check_vtx.Prop = MARK_TO_DELETE                      // Be sure and mark the convergent vertex for deletion
end function

function PROJ_VTX_TO_EDGE(vtx, edge, perp_dist, x_proj, y_proj, rp)
    edge_x_delta = edge.ToVtx.x - edge.FromVtx.x
    edge_y_delta = edge.ToVtx.y - edge.FromVtx.y
    edge_dist = SQRT(edge_x_delta * edge_x_delta + edge_y_delta * edge_y_delta)
    x_norm = edge_x_delta / edge_dist                    // Compute normalized values of line equation
    y_norm = edge_y_delta / edge_dist
    x_rel = vtx.x - edge.FromVtx.x
    y_rel = vtx.y - edge.FromVtx.y
    perp_dist = -(y_norm) * x_rel + x_norm * y_rel
    x_proj = x_rel + (perp_dist) * y_norm                // Compute relative x and y of projected point
    y_proj = y_rel + (perp_dist) * -(x_norm)
    if x_norm > y_norm &  x_norm <> 0.0
        rp = x_proj / x_norm * edge_dist                 // Position of the projected point along the edge (0 to 1.0)
    else if x_norm < y_norm & y_norm <> 0.0
        rp = y_proj / y_norm * edge_dist
    else
        rp = 0
        SET_STOP_ERROR                                   // Return rp = 0, degenerate edge detected, set  error and stop
    endif
    x_proj = x_proj + edge.FromVtx.x                     // Convert to absolute coordinates for the projected vertex
    y_proj = y_proj + edge.FromVtx.y
end function
```

```
function UPDATE_CONTOUR_LINE(root_vtx)
    vtx = root_vtx.NextVtx
    while vtx <> NULL
        if vtx.Prop = MARKED_FOR_DELETION    // Unlink the marked vertex
            vtx.PrevVtx.Next = vtx.NextVtx
            vtx.NextVtx.Prev = vtx.PrevVtx
        end if
        vtx = vtx..NextVtx
    end while
end function
```