**Large Crater Cluster Analysis Tool Version 2.0.** A. B. Author1 and C. D. Author2, 1Affiliation (include full mailing address and e-mail address if desired) for first author, 2Affiliation for second author (full mailing address and e-mail address).

**Introduction:** LCC Analysis Tools is a Geographic Information System (GIS) based tool suite that utilizes the spatial distribution (distance and density) of digitized secondary impact craters to assist in the identification of primary source craters. The development of this tool within a GIS fulfills two requirements: (1) analysis is repeatable and workflows easily documented and (2) the framework of a GIS provides the tools to work with geospatial data without additional overhead or abstraction, i.e. the source data and basemaps used to generate the intitial inputs to LCC can interact with the iterative output from LCC without additional processing.

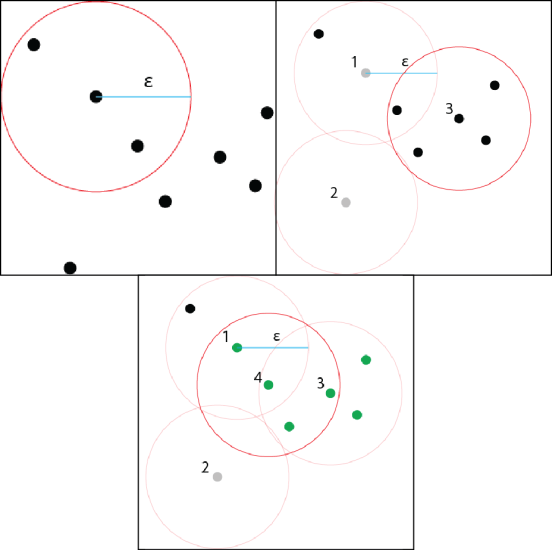
**Technical Methods:** The approximation of source craters requires that the secondary material ejection point be computed via the spatial distribution of a seconardy impacts. This section provides the technical background describing the proposed workflow and statistical methods implemented to fulfill this goal.

*Background.* Originally introduced by [1] LCC was designed as a plug-in to ArcMap 9.3 and underwent two revisions[2,3] to address bug fixes and introduce improved functionality. Major improvements included more robust clustering with large datasets, computation of trajectories accounting for the Corolis Effect, and the addition of extended log files for process tracking. With the release of ArcMap 10.0, an upgrade of the tool was required. This upgrade provided the initiative to explore additional spatial analysis techniques to potentially improve tool functionality. These include the removal of redundant processing phases, the addition of a new clustering technique, paramaterization of additional mopdel inputs, addition of optimization methods to guide initial tool usage, and weighting of trajectory point intersections.

*Technical Description.* LCC consists of five tools that can be used in either an iterative work environment, where parameters at each processing phase are explored and results analyzed, or a single end-to-end processing accepting the default or optimized parameters. Sequentially these tools are: Distance Matrix Computation, Clustering, Directional Distribution, Trajectory Extension, and Primary Impact Approximation.

*(1) Distance Matrix Computation.* A distance matrix is a dense matrix that stores the distance between all observations. It is diagonally symmetrical and therefore need only be computed for the upper right triangle. The computation of a distance matrix is O(n^2) and the storage of said matrix is O(n^2). [4] show that, for sufficiently large samples, it is not possible to compute a full distance matrix in a reasonable amount of time or store said matrix in RAM. Because the distance between each observation and *n* neighbors is an essential component to the next phases, we precompute the distance matrix and store it as a truncate, write to disk, distance table. In this way we store *k*-nearest neighbors to each observation (*n \* k*) total rows.

*(2) Clustering.* The clustering of seconadries into statistically meaningful groupings that potentially point back towards a source crater is the key component of this tool. With the version 2.0 release of LCC we have introduced a new density based clustering algorithm, DBScan [5,6]. DBScan introduces the notion of point pattern density to the computation of clusters. As seen in figure 1 Users need only define a minimum cluster size and minimum threshold distance. From this point the algorithm computes clusters that attain the necessary density and fulfill the distance constraint. While we do not suggest that this clustering method is a provable ideal technique for secondaries, it is extremely robust in two ways. First, the spatial distribution and shape of the underlying clusters do not need to fall within some predetermined range to be identified, i.e. a circular cluster of size *X* and a crescent cluster of size *Y* are both identifiable. Second, DBScan is computational efficient and provides speed improvements on the order of 6 times over the previsouly implemented hierarchal methods.



**Figure 1** – Example DBScan where ε is the threshold distance beyond which points are not clustered. (a) the point is check to see if sufficient points fall within ε. (b) a point *n* less than the threshold size is marked as noise. (c) a point previously marked as noise can be assigned to a cluster if a member of another points neighborhood that exceeds *n*

*(3) Directional Distribution*. The directionality and ellipticity of each statistically identified clusters provide metric by which a likely trajectory and trajectory confidence can be computed. We determine trajectory directionality by fitting each cluster with a bounding ellipse (to a user defined number of standard deviations). The trajectory is assume to bisect the centroid of the ellipse roughly along the semi-major axis (the error in bisection along the semi-major axis is a function of the length of the semi-major axis). The ellipticity or inverse flattening is defined using:

where *f* is equal to

where *a* is the semi-major axis and *b* is the semi-minor axis. Using this metric, ideal inverse flattening values are cloaser to 1.

*(4) Trajectory Extension*. Using the directional distribution of each cluster we then extend the semi-major axis along the geodesic to a user defined distance. Two major alterations to this tool have occurred with the upgrade to version 2.0. First, it is now possible to begin using the tool at this processing phase. For example, a user could digitize a series of linear crater chains using a best fit line and begin processing from here. Second, it is now possible to aggregate multiple datasets (an in-memory merge operation). We expect that the later functionality will allow users to compute clusters at various scale using the previous processing step and then perform a trajectory analysis that captures small and large scale crater clusters.

*(5) Primary Impact Approximation*. This tool seeks to locate a primary impact crater by clustering the intersection of multiple trajectories. [3] show that it is possible to utilize clusters of intersection points to compute the location of a potential primary. This version of the tool introduces the ability to weight the trajectory intersection based on the inverse product of the flattening metrics using:

where *l*  is a trajectory. Therefore the ideal intersection of two ideal trajectories will be 1. When a series of intersections are clustered, the centroid of that cluster is ‘pulled’ towards those intersections with weights closer to 1. Using the Zunil test case we see a 600m to 1000m offset using centroid weighting.

*Optimization*. Given the complexity of potential inputs we have adopted a model used in the ArcMap Geostatistical Analysis Tools; we provide the user with the option to populate relevant fields with ‘best guess’ optimized parameters to initiate processing. The addition of the optimization options provide the user with the ability to use this tool as a black box. In its current iteration, the tool optimized parameters using traditional descriptive statistics and generally seeks to either include a large portion of the data (less than or equal to the populate mean or mean plus standard deviation) or reduce overall dataset size (using those samples with values less than or equal to the lower quartile of the total population).

Iterative testing across a range of datasets is required to further tune the optimization and prodive an initial parameter space that fits the majority of use cases.

**Workflow:** We propose an iterative workflow with analysis of each intermediate step providing the rationale behing parameter selection in each subsequent step. For example, a distance table with *k* = 10 could be generated for repeated use in subsequent processing steps. Next both small and large scale cluster analysis could be performed by varying the noise threshold distance and minimum cluster distance parameters. Analysis of these results could drive additional clustering to identify a population of visually ‘ideal’ clusters. Then the directionality and ellipticity of said clusters would provide a means to winnow the data set prior to primary impact approximation.

**Availability:** LCC is available as an ArcMap Add-in via XXXXX. This distribution model requires a non-administrator installation. Once installed LCC is a dockable toolbar consisting of the five aforementioned processing phases.

**Future Work:** Future work will focus on the tuning of the tool to provide better best guess optimization. The realization of this goal requires that the tool be used on large and small crater clusters with diverse spatial distributions. Additionally, the tool will be applied to other stratographic features in order to assess the portability of the analytical techniques beyong crater clusters.

**References:** [1] Nava LPSC 2009, [2] Nava LPSC 2010, [3] Skinner LPSC 2011, [4] Laura Springer in review, [5] Ester et. al (1996) A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. [6] Sander et. al (1998) Density-Based Clustering in Spatial Databases.