Ground filtering with TIN refinement

We will now discuss an effective ground filtering method that is based on the *greedy* insertion of ground points into a TIN. Indeed, the same algorithmic paradigm of iterative TIN refinement that we saw earlier in Section 8.3 is used. The algorithm consists of three main steps:

1. construction of a rudimentary initial TIN (usually a Delaunay TIN);

2. computation of two geometric properties for each point that is not already labelled as ground;

3. incremental insertion of points that pass a simple and local 'ground test' based on the computed geometric properties.

The latter two steps are repeated until all remaining points fail the ground test.

In the first step a rudimentary initial TIN is constructed from a number of points that have locally the lowest elevation and are spread somewhat evenly over the data extent. These points are found by superimposing a 2D grid over the data extent and by selecting the lowest point for each grid cell (similar to grid thinning). The cell-size of the grid should be chosen such that it is larger than the largest non- ground object (usually a building). Thus, if the largest building has a footprint of 100x100m, the cell size should be a bit larger, eg 110m, so that it is guaranteed that each grid-cell has at least a few ground points. Each point that is inserted into the TIN is considered to be a ground point.

In the second step two geometric properties are computed for each unclassified point. These properties are based on the relation between the point *p* and the triangle in the current TIN that intersects its vertical projection. The two properties are illustrated in Figure 12.8a. The first property, denoted *d*, is the perpendicular distance between the *p* and the triangle. The second property, denoted α, is the largest angle of the angles between the triangle and the three vectors that connect each vertex with *p*.

In the ground test of the final step, it is simply checked for each point if its *d* is below a given threshold d_{max} and if its α is below a given threshold α_{max}. If this is indeed the case, the point is labelled as a ground point and inserted into the TIN. Compare Figures 12.8b and 12.8c.

Of course, if the triangles in the TIN change, the properties of the overlapping unclassified points need to be recomputed. However, the algorithm is greedy, which means that it never "goes back" on operations that were previously performed, and thus when a point *p* is inserted as the ground, it is never removed. When all remaining points fail the ground test, the algorithm terminates. Figure 12.9 gives an example result.