Triangular Plane Mesh Generation: van der Corput sequence? [1]

Jingjin Yu

Plane mesh generation with *fat* triangles, due to its importance toward numerically solving certain classes of partial differential equations (PDEs), has been an active research field for the last several decades [1]. The state-of-art approach applies the *quadtree* data structure to generate a non-uniform mesh that also contains comparatively small numbers of triangles. In this proposal, we raise this specific question: is there a data structure that we can use to give comparable or better mesh than the quadtree data structure?

In particular, we have in mind the *van de Corput sequence* in higher dimensions. van de Corput sequence [2] is a low-discrepancy sequence over the unit interval constructed by reversing the base n representation of the sequence of natural numbers. For example, the first 9 numbers of the binary van de Corput sequence are:

$$0.1_2, 0.01_2, 0.11_2, 0.001_2, 0.101_2, 0.011_2, 0.111_2, 0.0001_2, 0.1001_2, \ldots,$$

or, equivalently,

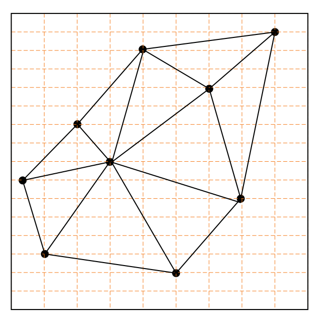
$$\frac{1}{2}, \frac{1}{4}, \frac{3}{4}, \frac{1}{8}, \frac{5}{8}, \frac{3}{8}, \frac{7}{8}, \frac{1}{16}, \frac{9}{16}, \ldots.$$

It is clear that any first k points in this binary sequence have relatively uniform neighboring distance; if we take some $k = 2^m - 1$, then the neighboring distance is uniform.

Extending the sequence to higher dimensions can be achieved by using sequences with different basis. For example, using base 2 and 3 for x and y on a one by one square, the first 9 points in 2D are:

$$(\frac{1}{2}, \frac{1}{3}), (\frac{1}{4}, \frac{2}{3}), (\frac{3}{4}, \frac{1}{9}), (\frac{1}{8}, \frac{4}{9}), (\frac{5}{8}, \frac{7}{9}), (\frac{3}{8}, \frac{2}{9}), (\frac{7}{8}, \frac{5}{9}), (\frac{1}{16}, \frac{8}{9}), (\frac{9}{16}, \frac{1}{27}), \ldots.$$

Drawing them on the square gives the points on the figure to the right, which yields a reasonable triangulation among themselves (Obviously, similar to the smoothing taught in class, moving the point $(\frac{1}{2}, \frac{1}{3})$ a little can improve on the angles). If we could maintain the boundaries to fit the input points well and grow the point set only when it is necessary, a reasonable triangular mesh is possible. Moreover, we want to consider whether it is possible to avoid directly applying Delaunay refinement in the process. We will not limit ourselves to only van de Corput sequences; other deterministic, incremental sequences will also be considered.



The quadtree is a top-down approach in the sense that it subdivides the squares into smaller ones. The algorithm by Shewchuk appears to have some of this flavor and some of the bottom-up flavor. What we are trying to do is similar to that of Shewchuk's algorithm. It would be also interesting to study the possibility of a bottom-up approach that grow the point set along the boundaries.

[1]We omit the obvious references due to the 1-page space limit.

[2]Most of this intro on 1D van de Corput sequence is borrowed from wiki.